The Challenges of Extreme Scale Computing

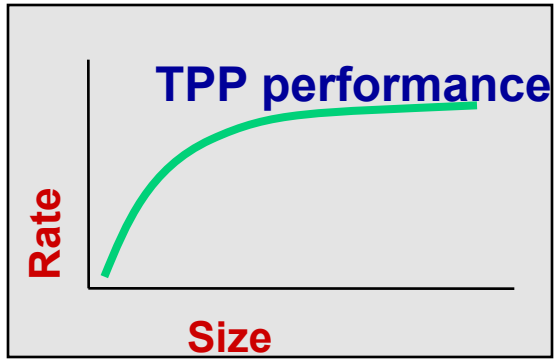
Jack Dongarra

University of Tennessee
Oak Ridge National Laboratory
University of Manchester

H. Meuer, H. Simon, E. Strohmaier, & JD

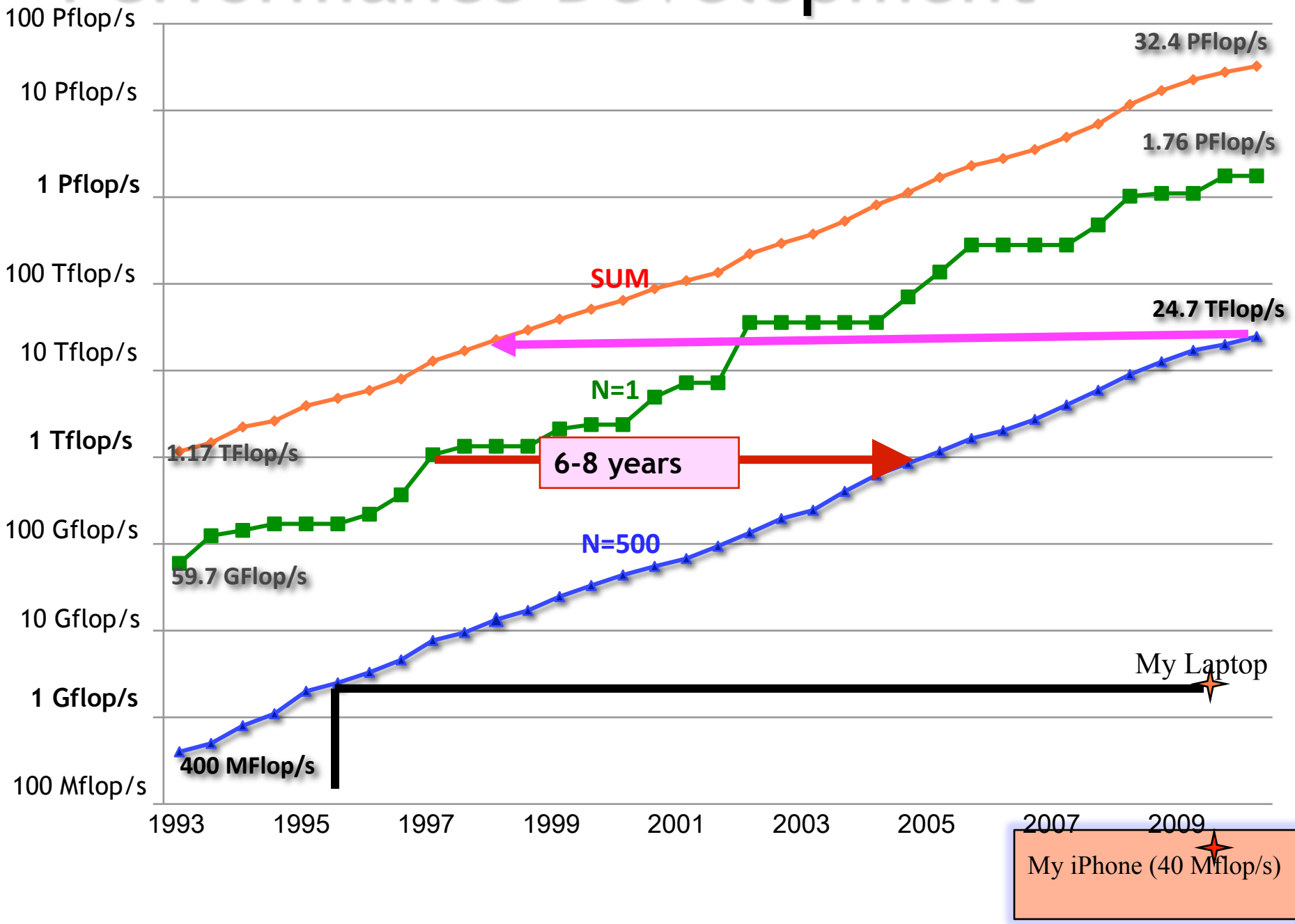
- Listing of the 500 most powerful Computers in the World
- Yardstick: Rmax from LINPACK MPP

$Ax=b$, *dense problem*



- Updated twice a year
  - SC‘xy in the States in November
  - Meeting in Germany in June
- All data available from **www.top500.org**

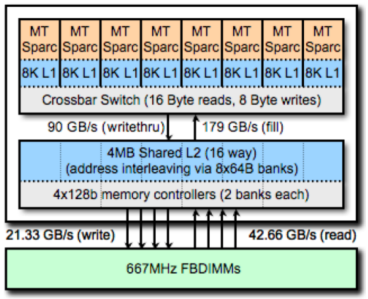
Performance Development



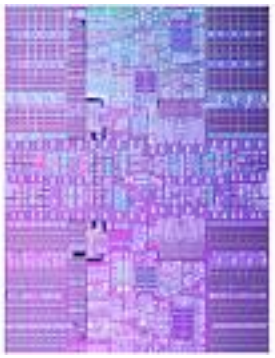
Today's Multicores

99% of Top500 Systems Are Based on Multicore

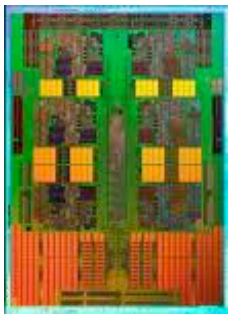
Of the Top500,
499 are multicore.



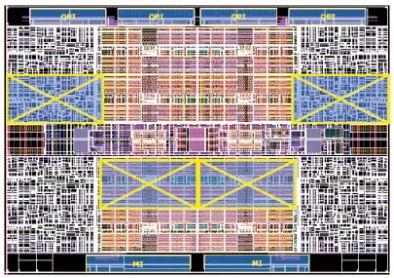
Sun Niagra2 (8 cores)



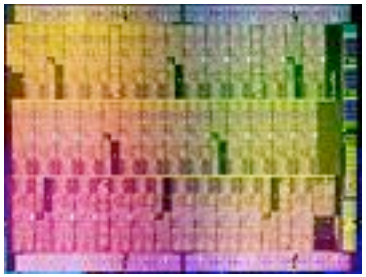
IBM Power 7 (8 cores)



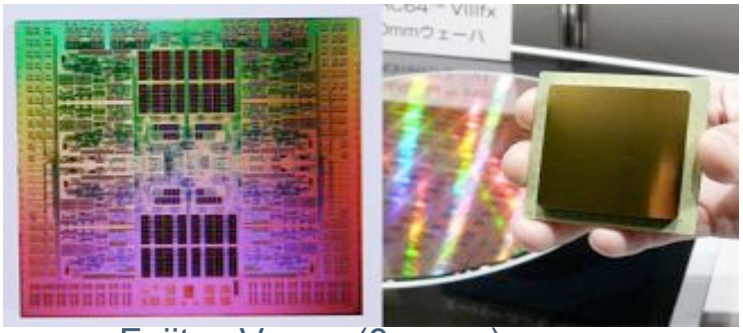
AMD Magny Cours
(12 cores)



Intel Xeon(8 cores)



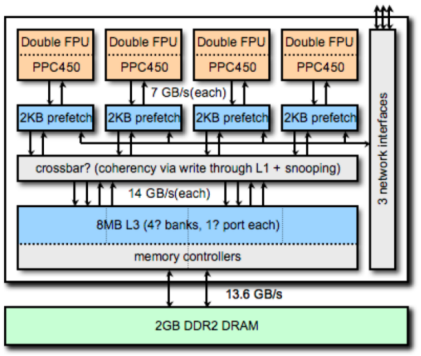
Intel Knight's Corner
(40 cores)



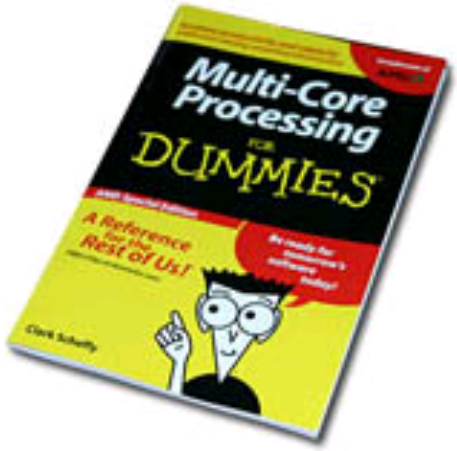
Fujitsu Venus (8 cores)

Processors in Top500:

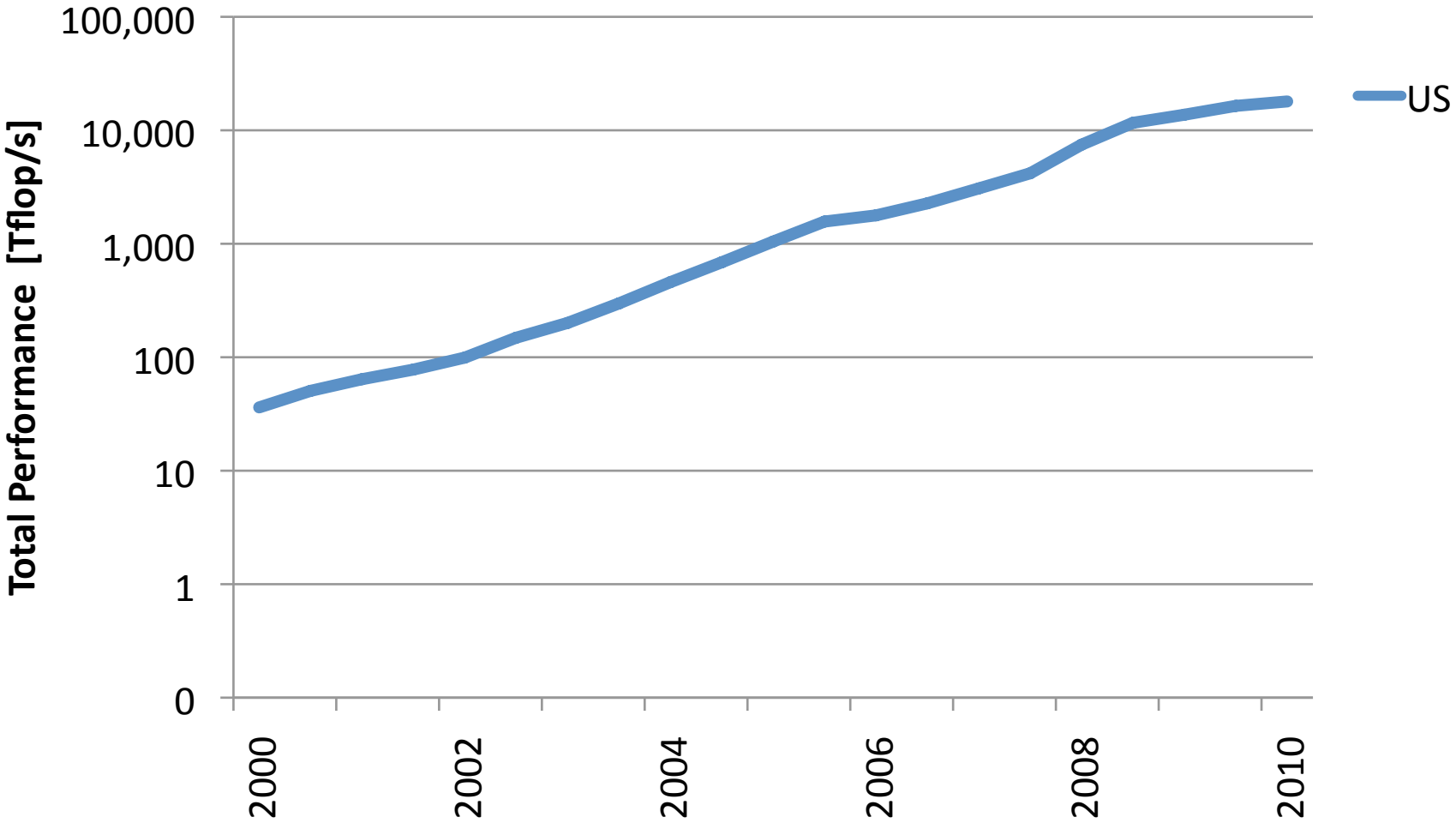
Intel 81%
AMD 10%
IBM 8%



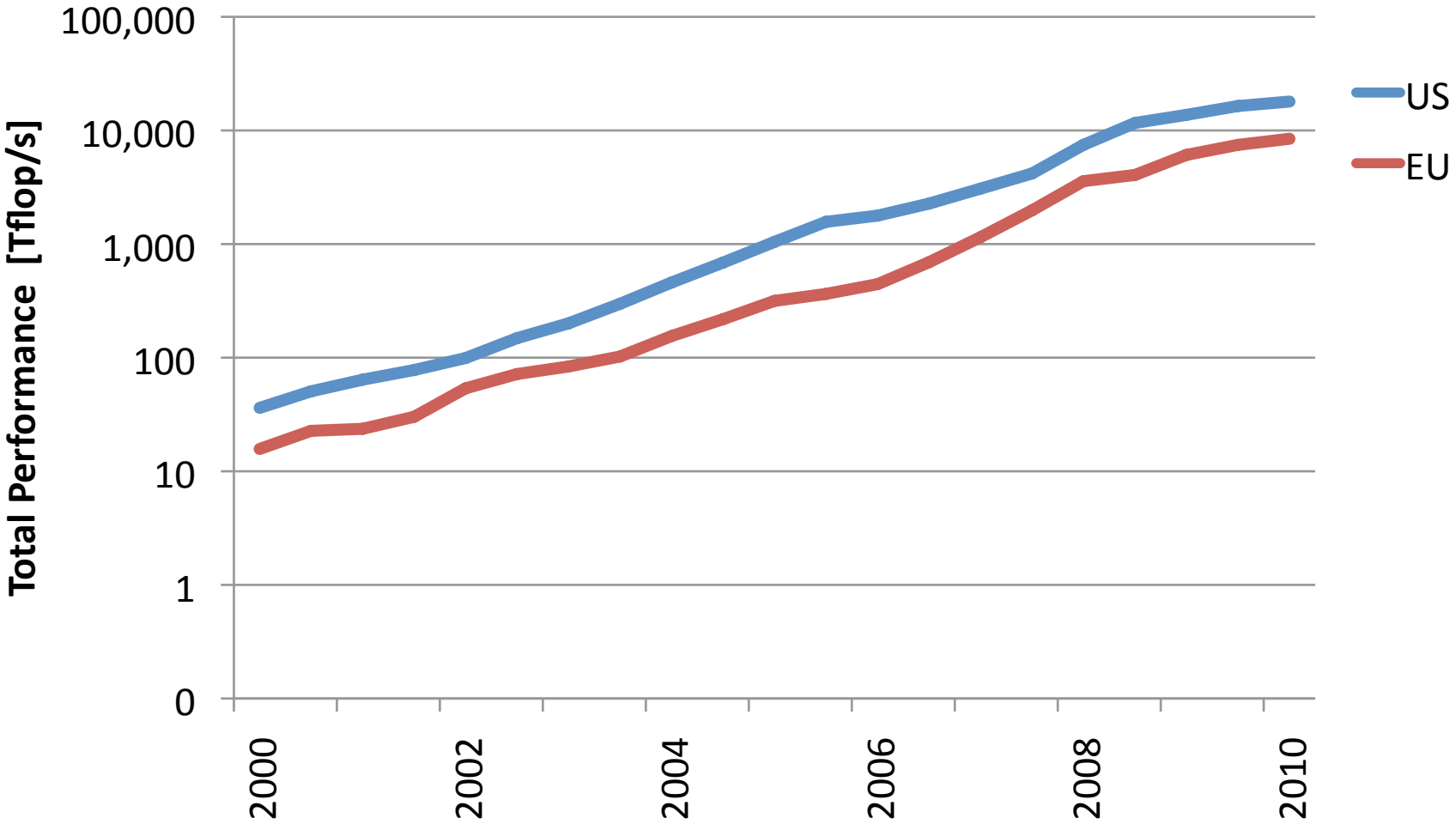
IBM BG/P (4 cores)



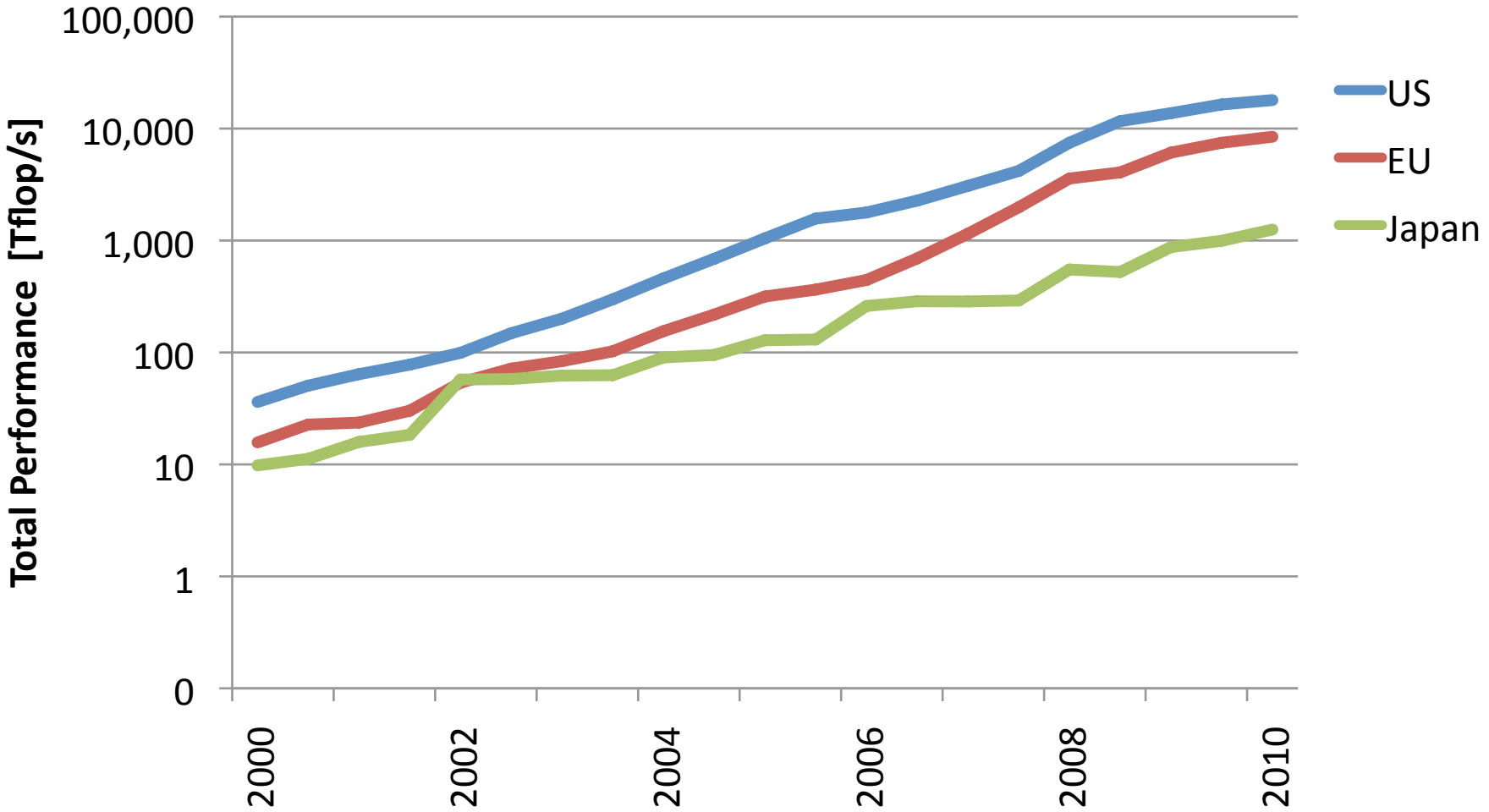
Performance of Countries



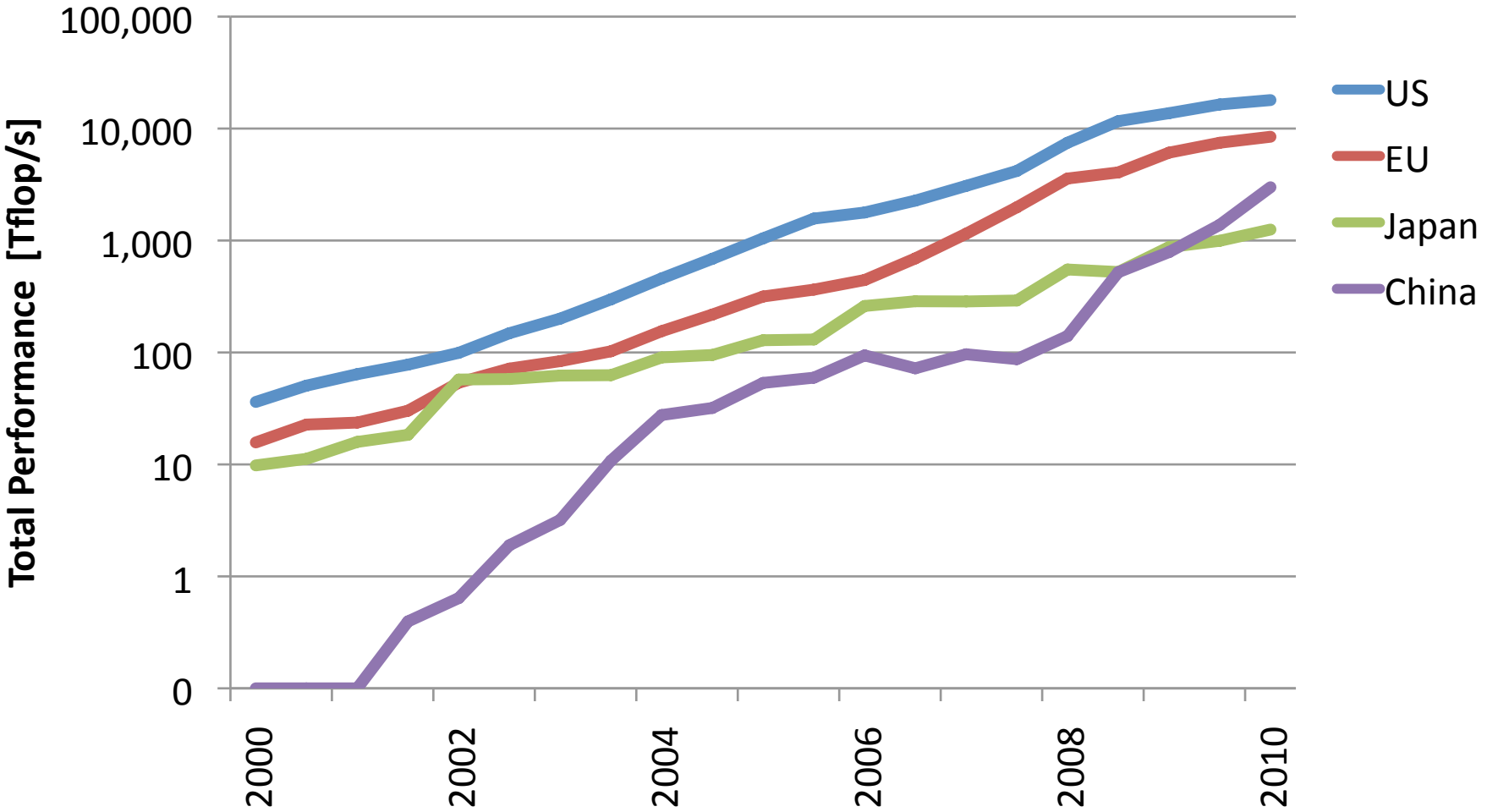
Performance of Countries



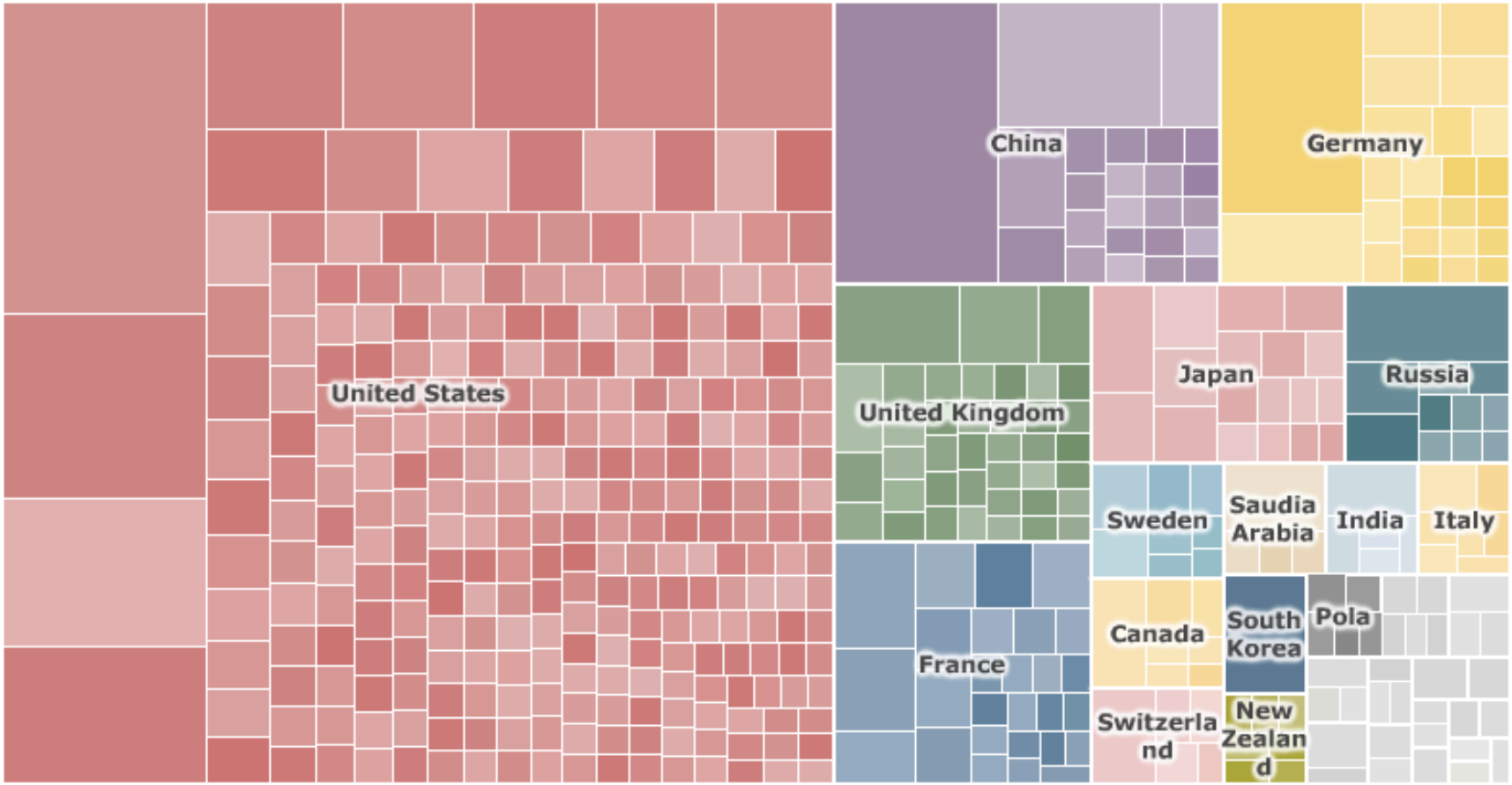
Performance of Countries



Performance of Countries



Countries / System Share

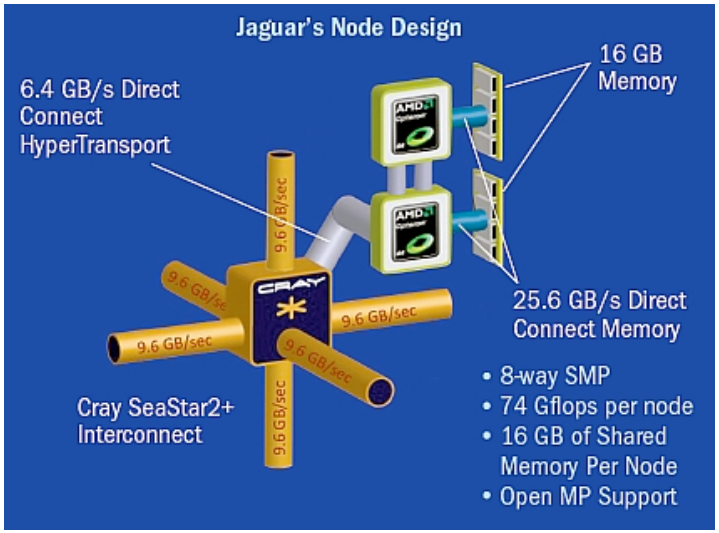


24 systems in the Germany

#1 ORNL's Newest System Jaguar XT5

2.3 Pflop/s system with more than 224K cores using AMD's 6 Core chip.

Peak performance	**2.332 PF**
System memory	300 TB
Disk space	10 PB
Disk bandwidth	240+ GB/s
Interconnect bandwidth	374 TB/s



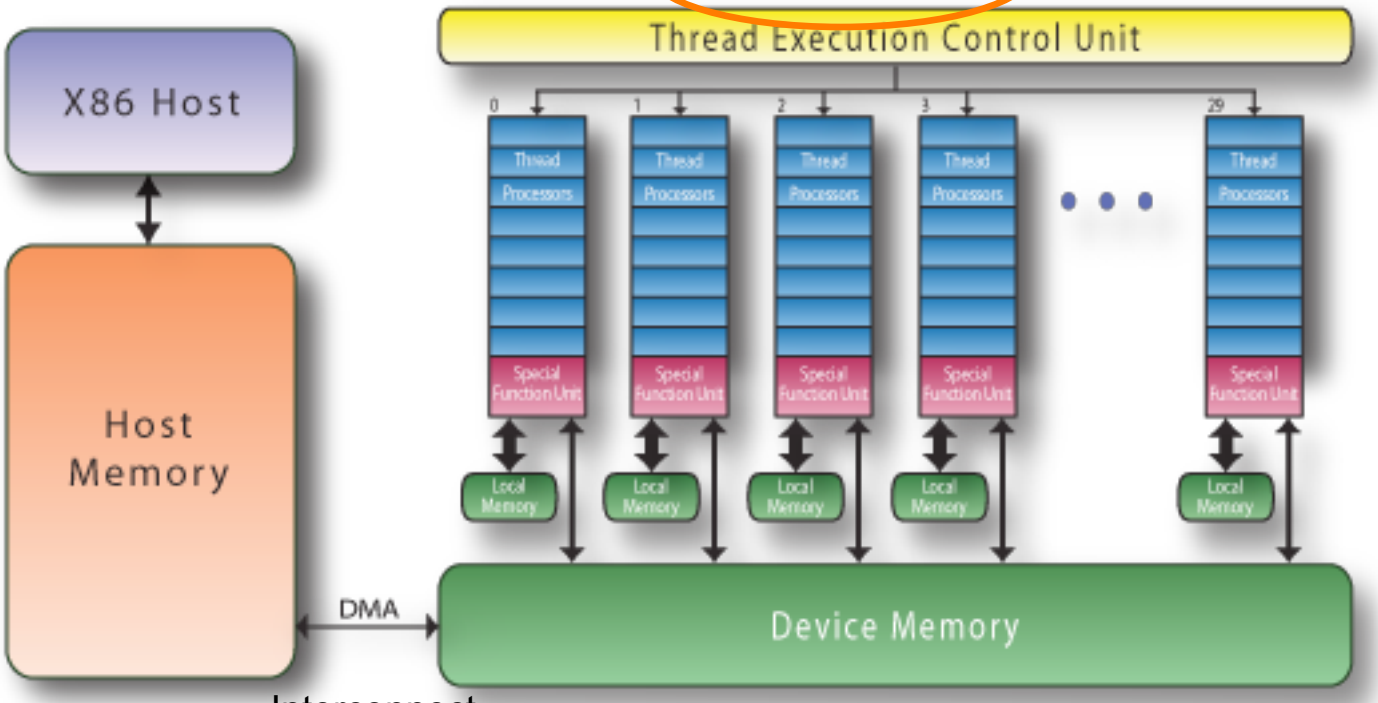
U.S. DEPARTMENT OF ENERGY | Office of Science

#2 – National Supercomputer Center in Shenzhen, China – Dawning Integrator

- Nebulae
- Hybrid system, commodity + GPUs
- Theoretical peak 2.98 Pflop/s
- Linpack Benchmark at 1.27 Pflop/s
- 4640 nodes, each node:
  2 Intel 6-core Xeon5650 + Nvidia Fermi C2050 GPU (each 14 co
  - 120,640 cores
  - Infiniband connected

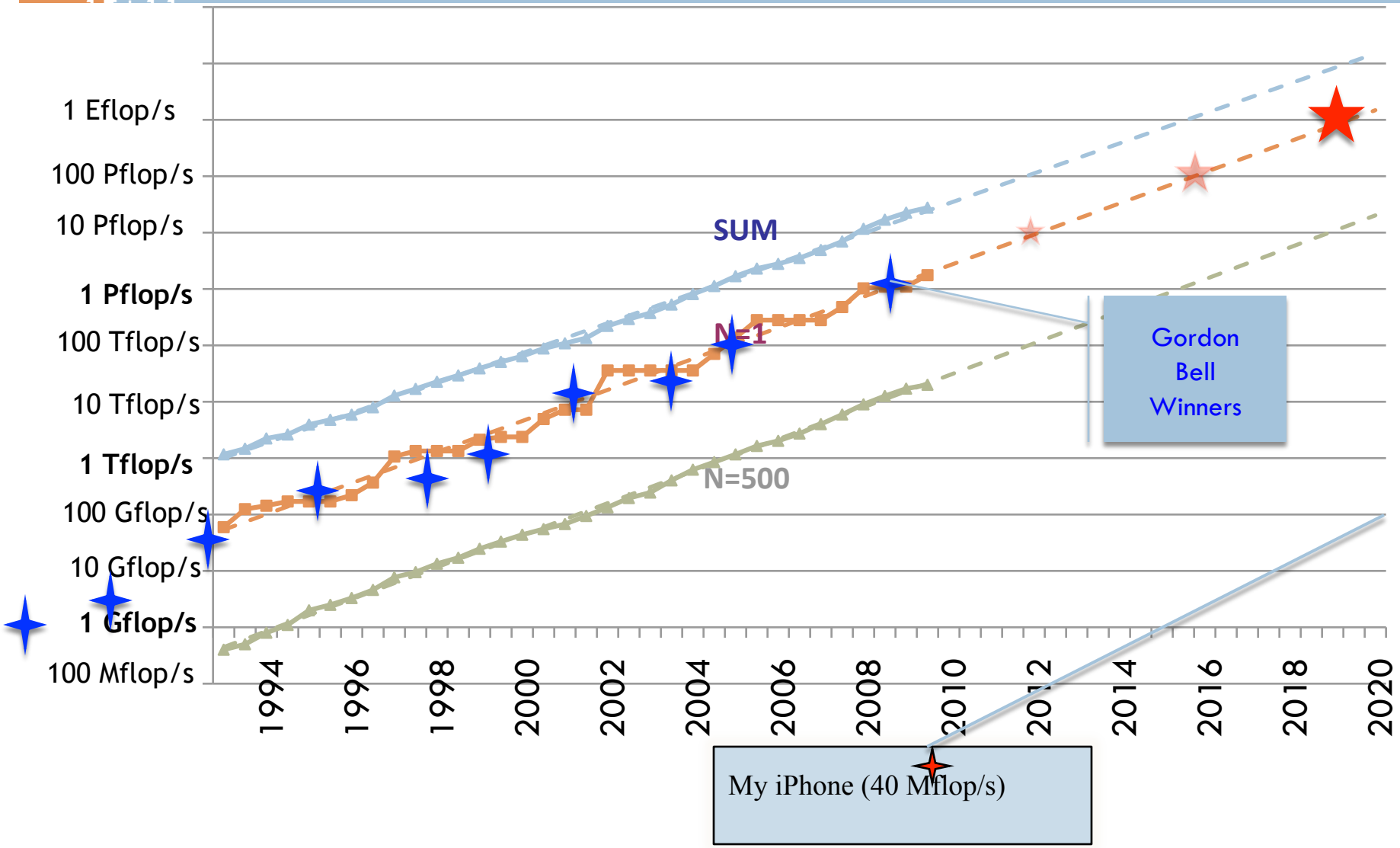
Commodity plus Accelerators

Commodity	Accelerator (GPU)
Intel Xeon	Nvidia C2050 “Fermi”
8 cores	448 “Cuda cores”
3 GHz	1.15 GHz
8*4 ops/cycle	~~448 ops/cycle~~
96 Gflop/s (DP)	515 Gflop/s (DP)



Interconnect
PCI Express
512 MB/s to 32GB/s
8 MW – 512 MW

Performance Development in Top500



Potential System Architecture with a cap of $200M and 20MW

Systems	2010
System peak	**2 Pflop/s**
Power	**6 MW**
System memory	0.3 PB
Node performance	125 GF
Node memory BW	25 GB/s
Node concurrency	12
Total Node Interconnect BW	3.5 GB/s
System size (nodes)	18,700
Total concurrency	225,000
Storage	15 PB
IO	0.2 TB
MTTI	days

Potential System Architecture with a cap of $200M and 20MW

Systems	2010	2018
System peak	**2 Pflop/s**	**1 Eflop/s**
Power	**6 MW**	**~20 MW**
System memory	0.3 PB	32 - 64 PB **[.03 Bytes/Flop]**
Node performance	125 GF	1,2 or 15TF
Node memory BW	25 GB/s	2 - 4TB/s **[.002 Bytes/Flop]**
Node concurrency	12	O(1k) or 10k
Total Node Interconnect BW	3.5 GB/s	200-400GB/s (1:4 or 1:8 from memory BW)
System size (nodes)	18,700	O(100,000) or O(1M)
Total concurrency	225,000	O(billion)
Storage	15 PB	500-1000 PB (>10x system memory is min)
IO	0.2 TB	60 TB/s (how long to drain the machine)
MTTI	days	O(1 day)

Potential System Architecture with a cap of $200M and 20MW

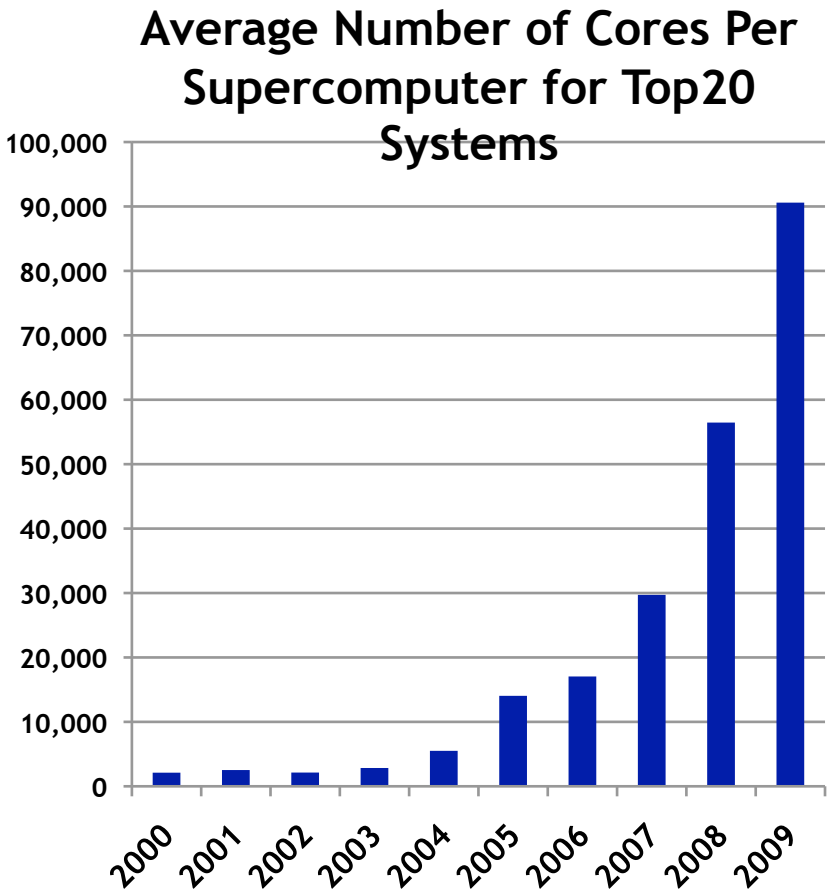
Systems	2010	2018	Difference Today & 2018
System peak	**2 Pflop/s**	**1 Eflop/s**	**O(1000)**
Power	**6 MW**	**~20 MW**	
System memory	0.3 PB	32 - 64 PB **[.03 Bytes/Flop]**	O(100)
Node performance	125 GF	1,2 or 15TF	O(10) - O(100)
Node memory BW	25 GB/s	2 - 4TB/s **[.002 Bytes/Flop]**	O(100)
Node concurrency	12	O(1k) or 10k	O(100) - O(1000)
Total Node Interconnect BW	3.5 GB/s	200-400GB/s (1:4 or 1:8 from memory BW)	O(100)
System size (nodes)	18,700	O(100,000) or O(1M)	O(10) - O(100)
Total concurrency	225,000	O(billion)	O(10,000)
Storage	15 PB	500-1000 PB (>10x system memory is min)	O(10) - O(100)
IO	0.2 TB	60 TB/s (how long to drain the machine)	O(100)
MTTI	days	O(1 day)	- O(10)

Exascale (10^{18} Flop/s) Systems: Two possible paths

- **Light weight processors (think BG/P)**
  - ~1 GHz processor (10^9)
  - ~1 Kilo cores/socket (10^3)
  - ~1 Mega sockets/system (10^6)
- **Hybrid system (think GPU based)**
  - ~1 GHz processor (10^9)
  - ~10 Kilo FPUs/socket (10^4)
  - ~100 Kilo sockets/system (10^5)

Factors that Necessitate Redesign of Our Software

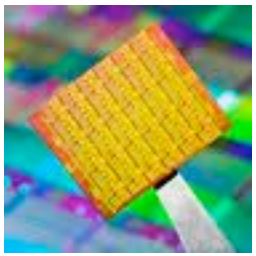
- Steepness of the ascent from terascale to petascale to exascale
- Extreme parallelism and hybrid design
  - Preparing for million/billion way parallelism
- Tightening memory/bandwidth bottleneck
  - Limits on power/clock speed implication on multicore
  - Reducing communication will become much more intense
  - Memory per core changes, byte-to-flop ratio will change
- Necessary Fault Tolerance
  - MTTF will drop
  - Checkpoint/restart has limitations



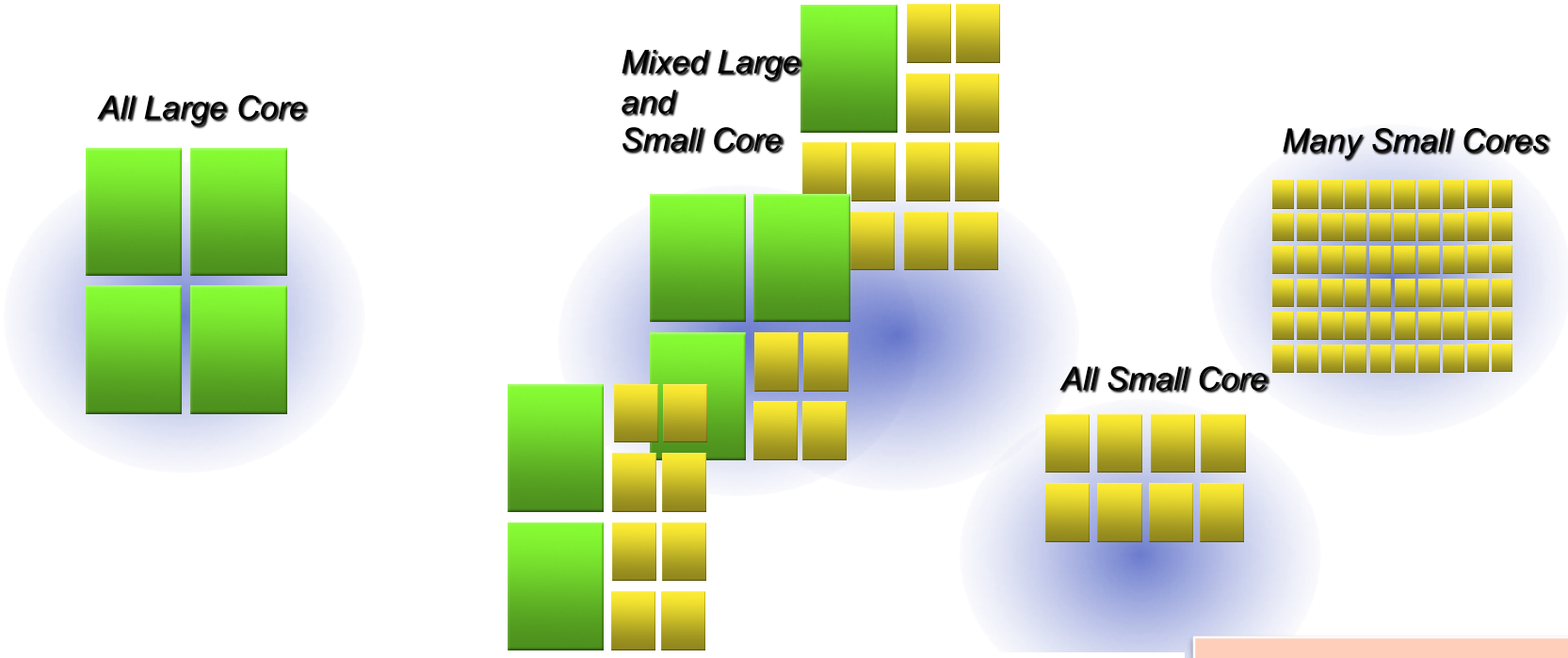
Software infrastructure does not exist today

Future Computer Systems

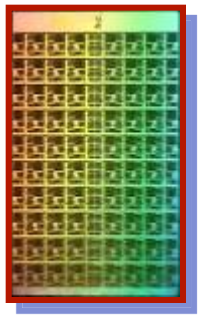
- Most likely be a hybrid design
- Think standard multicore chips and accelerator (GPUs)
- Today accelerators are attached
- Next generation more integrated
- Intel's Larrabee now called "Knights Corner" and "Knights Ferry" to come.
  - 48 x86 cores
- AMD's Fusion in 2011 - 2013
  - Multicore with embedded graphics ATI
- Nvidia's plans?

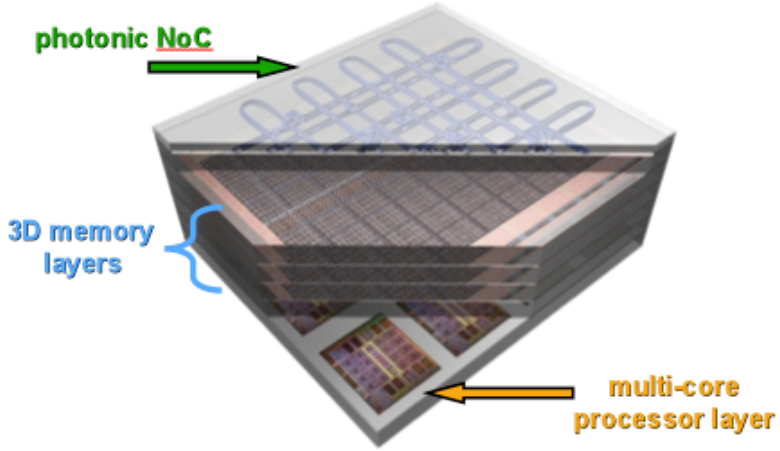


What's Next?



Many Floating-Point Cores





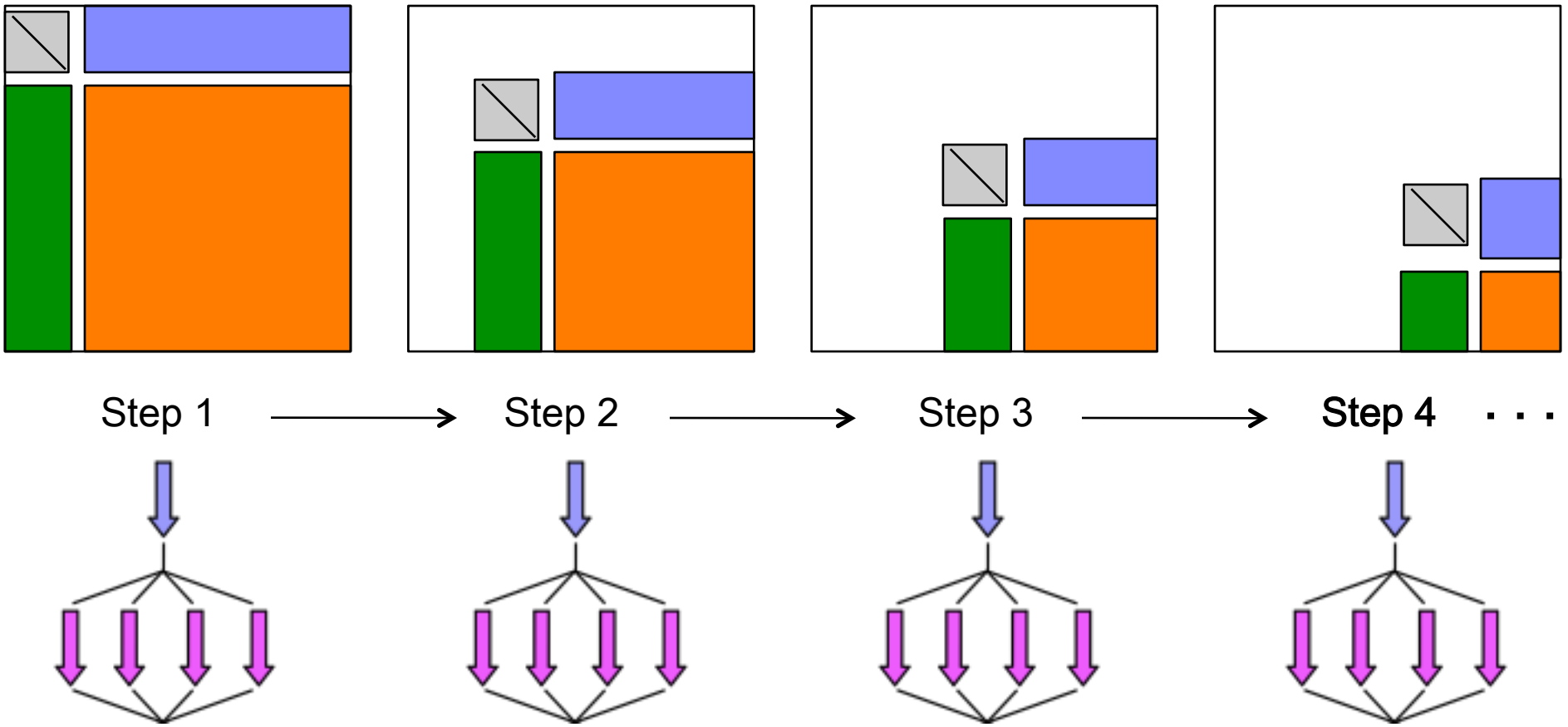
+ 3D Stacked Memory

Different Classes of Chips
- Home
- Games / Graphics
- Business
- Scientific

Major Changes to Software

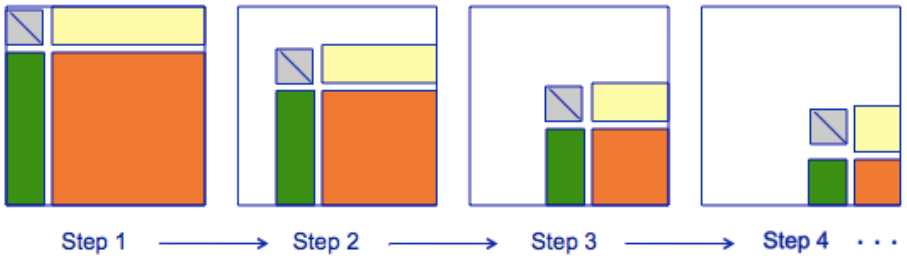
- Must rethink the design of our software
  - Another disruptive technology
    - Similar to what happened with cluster computing and message passing
  - Rethink and rewrite the applications, algorithms, and software
- Numerical libraries for example will change
  - For example, both LAPACK and ScaLAPACK will undergo major changes to accommodate this

LAPACK LU/LL^T/QR

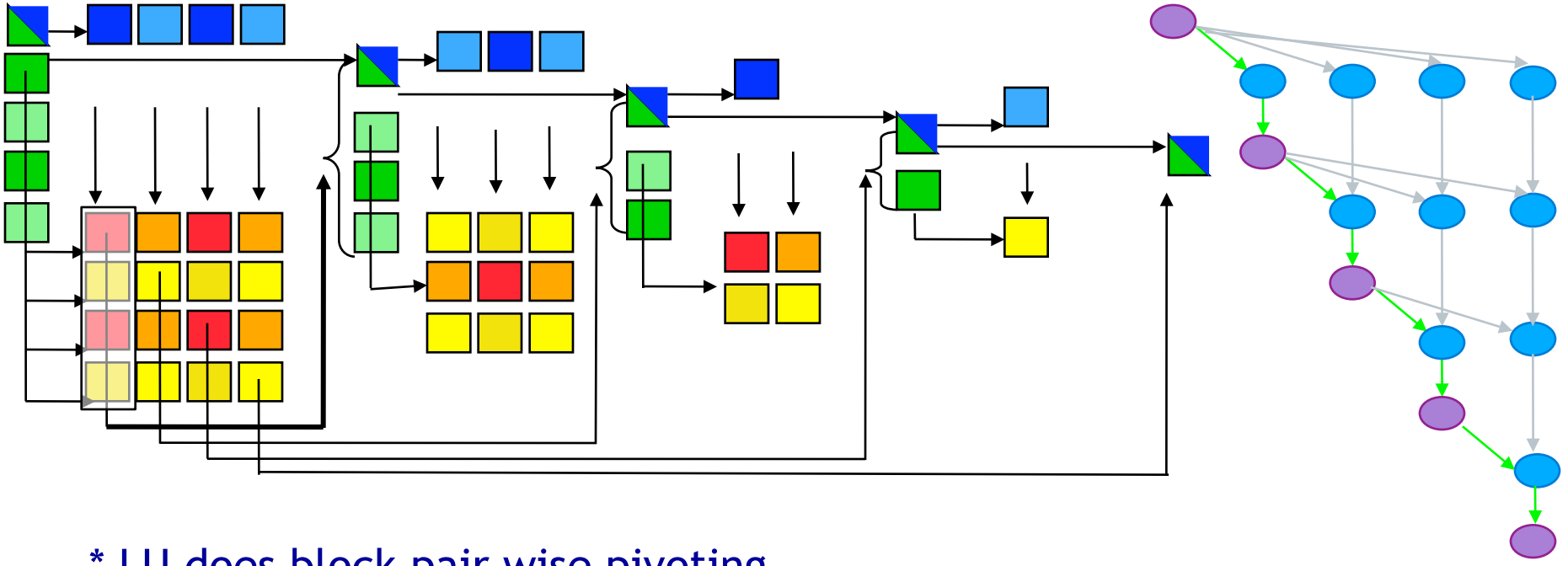


- **Fork-join, bulk synchronous processing**

Parallel Tasks in LU/LL^T/QR



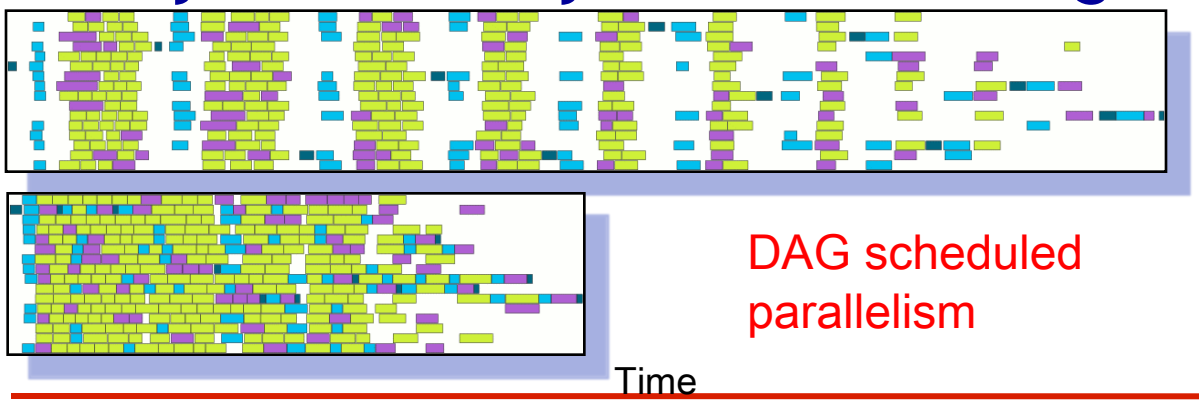
- Break into smaller tasks and remove dependencies

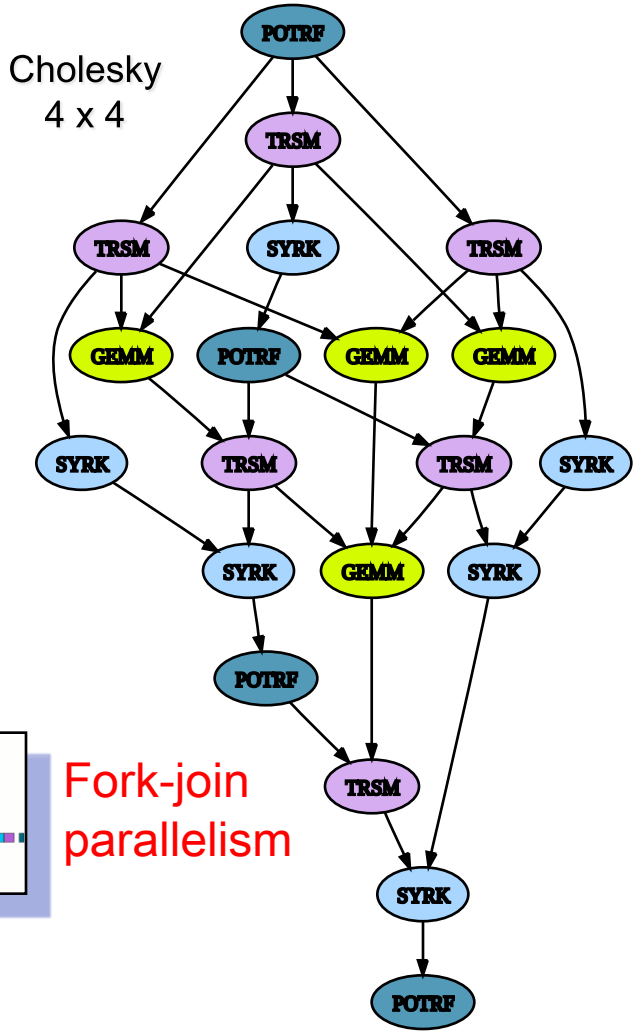


* LU does block pair wise pivoting

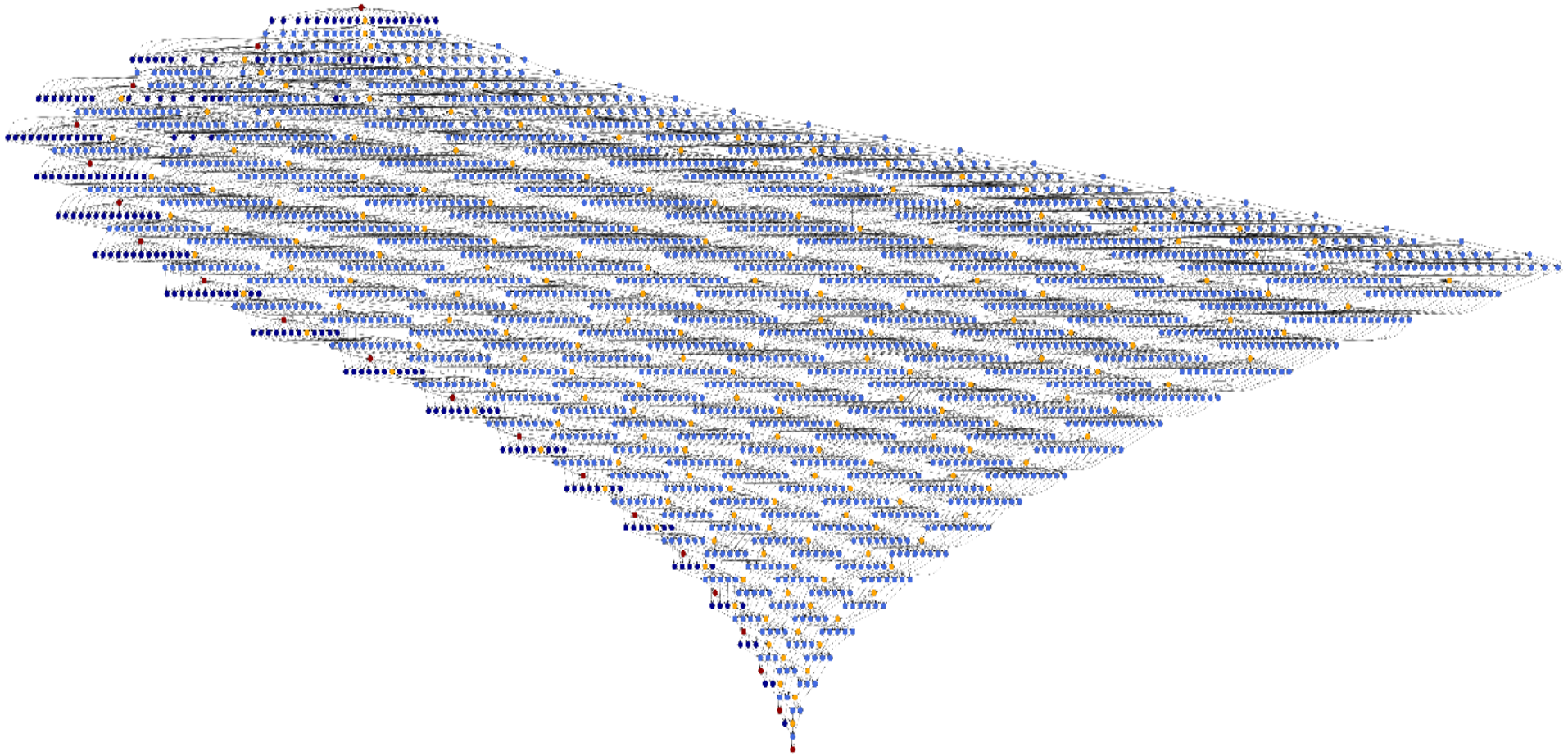
PLASMA: Parallel Linear Algebra s/w for Multicore Architectures

- **Objectives**
  - High utilization of each core
  - Scaling to large number of cores
  - Shared or distributed memory
- **Methodology**
  - Dynamic DAG scheduling
  - Explicit parallelism
  - Implicit communication
  - Fine granularity / block data layout
- **Arbitrary DAG with dynamic scheduling**





LU DAG representation

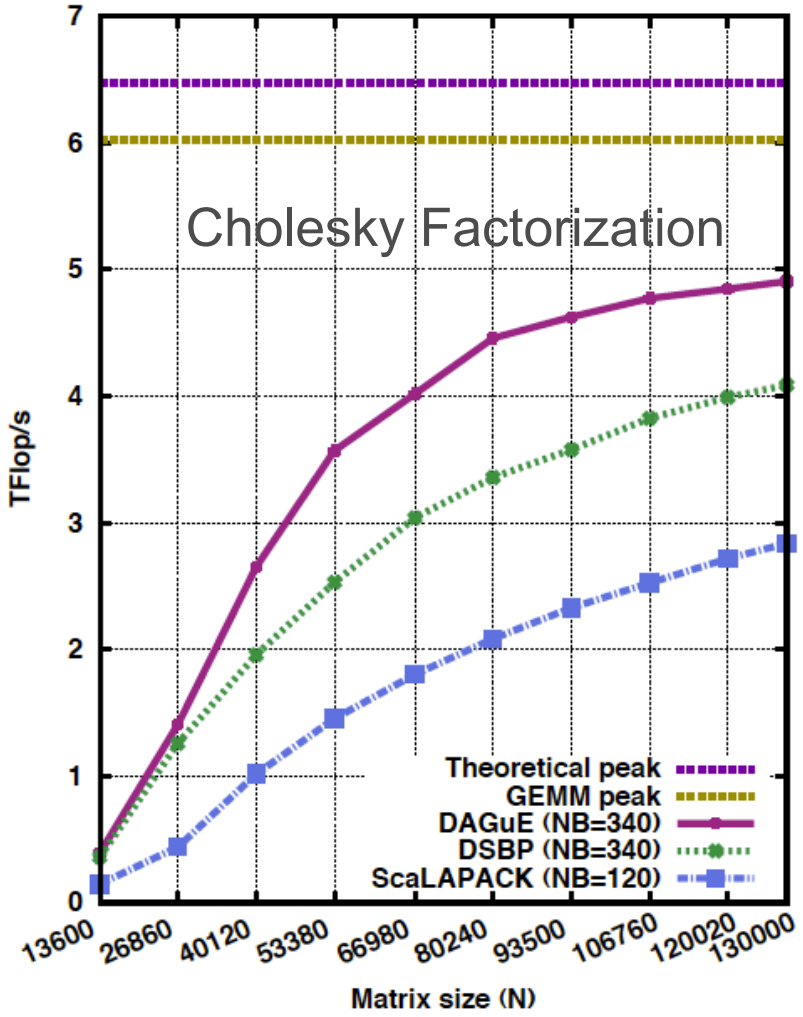


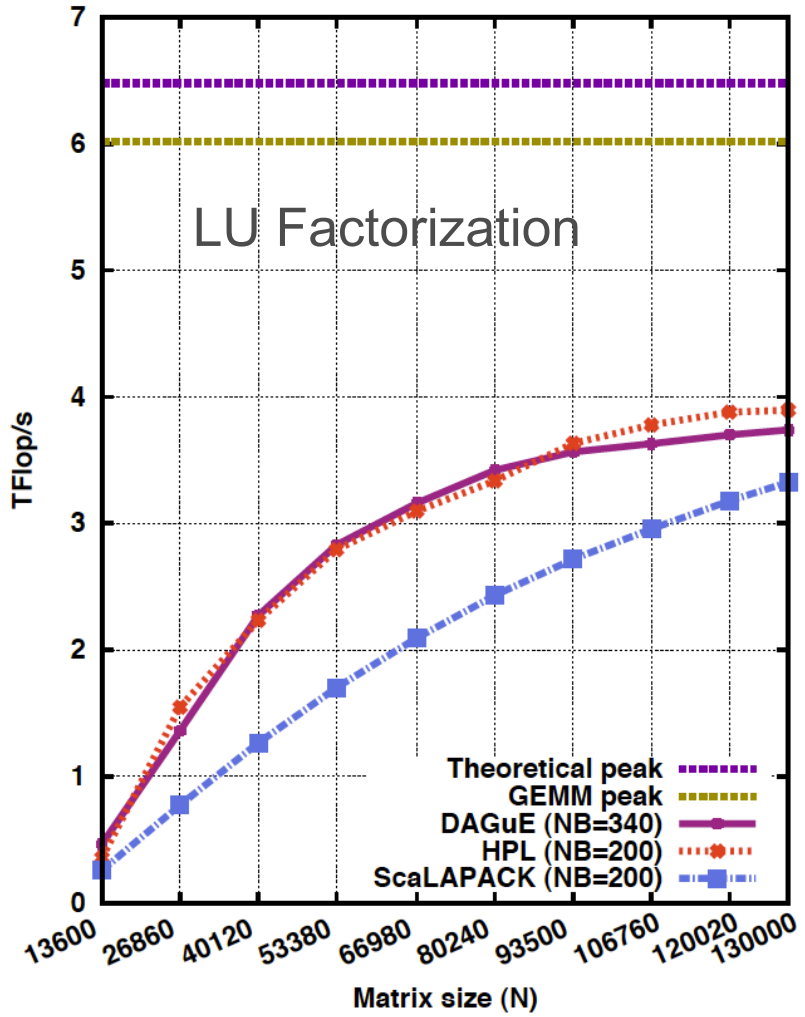
Can’t represent the whole DAG explicitly
Can’t have a single runtime process dispatch work

DPLASMA / DAGuE

- DAGuE: the runtime
  - Deploy a DAG on a heterogeneous distributed environment
  - Architecture aware
    - Minimize data movements (in and out the node)
    - Enforce data locality (cache / NUMA / GPU)
  - Move the data across nodes
- DPLASMA: a algebraic description of a DAG
  - Define the data distribution
  - Describe the algorithm at a high level

DPLASMA/DAGuE on Cholesky and LU Factorization





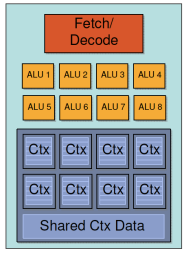
Griffon : 81 nodes, (dual socket, quad core Intel 2.5 GHz), 648 cores, Infiniband 20Gbs

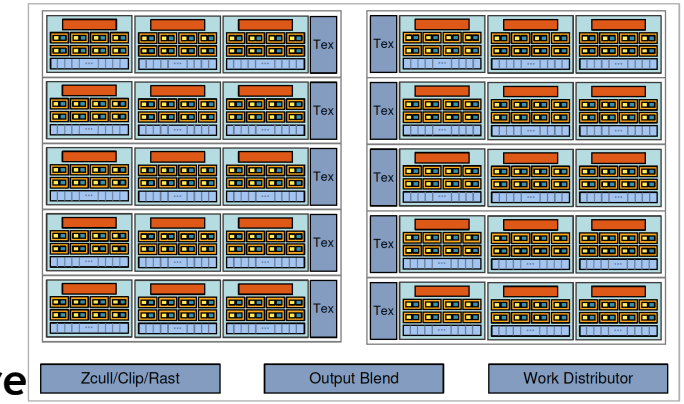
Challenges of using GPUs

- **High levels of parallelism**
  - **Many GPU cores, serial kernel execution**
  - [e.g. 240 in the Nvidia Tesla; up to 512 in *Fermi* - to have concurrent kernel execution]
- **Hybrid/heterogeneous architectures**
  - **Match algorithmic requirements to architectural strengths**
  - [e.g. small, non-parallelizable tasks to run on CPU, large and parallelizable on GPU]
- **Compute *vs* communication gap**
  - **Exponentially growing gap; persistent challenge**
  - [Processor speed improves 59%, memory bandwidth 23%, latency 5.5%]
  - [on all levels, e.g. a GPU Tesla C1070 (4 x C1060) has compute power of O(1,000) Gflop/s but GPUs communicate through the CPU using O(1) GB/s connection]

NVIDIA GeForce GTX 280 (Tesla C1060)

Processing Core





- **NVIDIA-Speak**
  - 240 CUDA cores (ALUs)
- **Generic speak**
  - 30 processing cores
    - 8 CUDA Cores (SIMD functional units) per core
  - 1 mul-add (2 flops) + 1 mul per functional unit (3 flops/cycle)
  - Best case theoretically: 240 mul-adds + 240 muls per cycle
    - 1.3 GHz clock
    - 30 * 8 * (2 + 1) * 1.33 = 933 Gflop/s peak
  - Best case reality: 240 mul-adds per clock
    - Just able to do the mul-add so 2/3 or 624 Gflop/s
  - All this is single precision
    - Double precision is 78 Gflop/s peak (Factor of 8 from SP; exploit mixed prec)
  - 141 GB/s bus, 1 GB memory
  - 4 GB/s via PCIe (we see: T = 11 us + Bytes/3.3 GB/s)
  - In SP SGEMM performance 375 Gflop/s

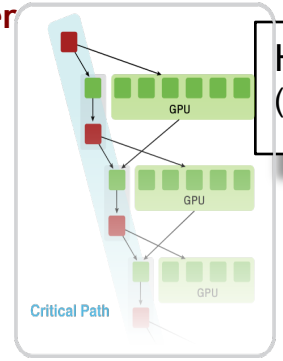
NVIDIA Tesla C2050 (Fermi), GF100 Chip

- **NVIDIA-Speak**
  - 448 CUDA cores (ALUs)
- **Generic speak**
  - 14 processing cores
    - 32 CUDA Cores (SIMD functional units) per core
  - 1 mul-add (2 flops) per ALU (2 flops/cycle)
  - Best case theoretically: 448 mul-adds
    - 1.15 GHz clock
    - 14 * 32 * 2 * 1.15 = 1.03 Tflop/s peak
  - All this is single precision
    - Double precision is half this rate, 515 Gflop/s
  - 144 GB/s bus, 3 GB memory
  - In SP SGEMM performance 580 Gflop/s
  - In DP DGEMM performance 320 Gflop/s
  - Power: 247 W
  - Interface PCIex16

Processing Core

Developing heterogeneous, multi -core-aware algorithms and software

- **Dense solvers for multicore/GPUs – MAGMA Project**
  - MAGMA - based on LAPACK and extended for hybrid systems (multi-GPUs + multicore systems);
  - MAGMA - designed to be similar to LAPACK in functionality, data storage and interface, to allow scientists to effortlessly port any LAPACK-relying software components to take advantage of new architectures
  - MAGMA - to leverage years of experience in developing open source LA software packages and systems like LAPACK, ScaLAPACK, BLAS, ATLAS as well as the newest LA developments (e.g. communication avoiding algorithms) and experiences on homogeneous multicores (e.g. PLASMA)
- **MAGMA uses HYBRIDIZATION methodology based on**
  - Representing linear algebra algorithms as collections of TASKS and DATA DEPENDENCIES among them
  - Properly SCHEDULING tasks' execution over multicore and GPU hardware components

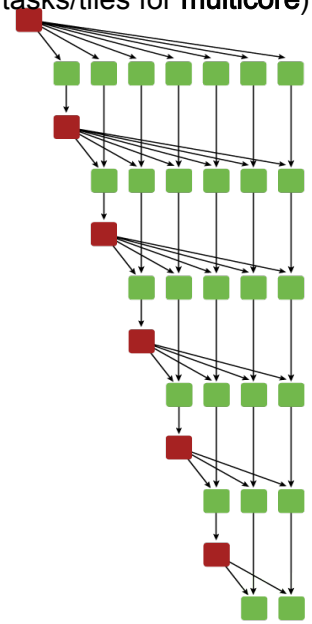


Hybrid CPU+GPU algorithms
(small tasks for multicores and large tasks for GPUs)

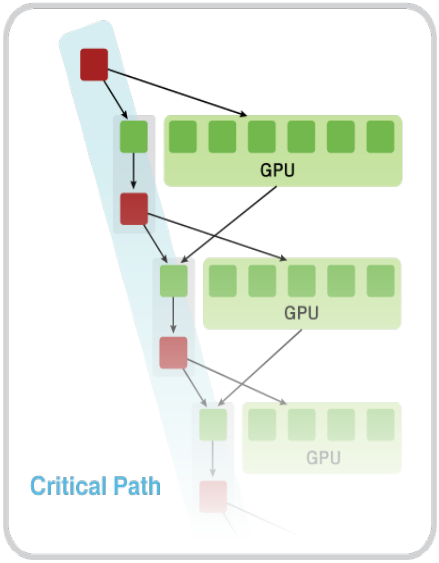
Hybrid Computing

- **Match algorithmic requirements to architectural strengths of the hybrid components**
  Multicore : small tasks/tiles
  Accelerator: large data parallel tasks

Algorithms as DAGs
(small tasks/tiles for **multicore**)

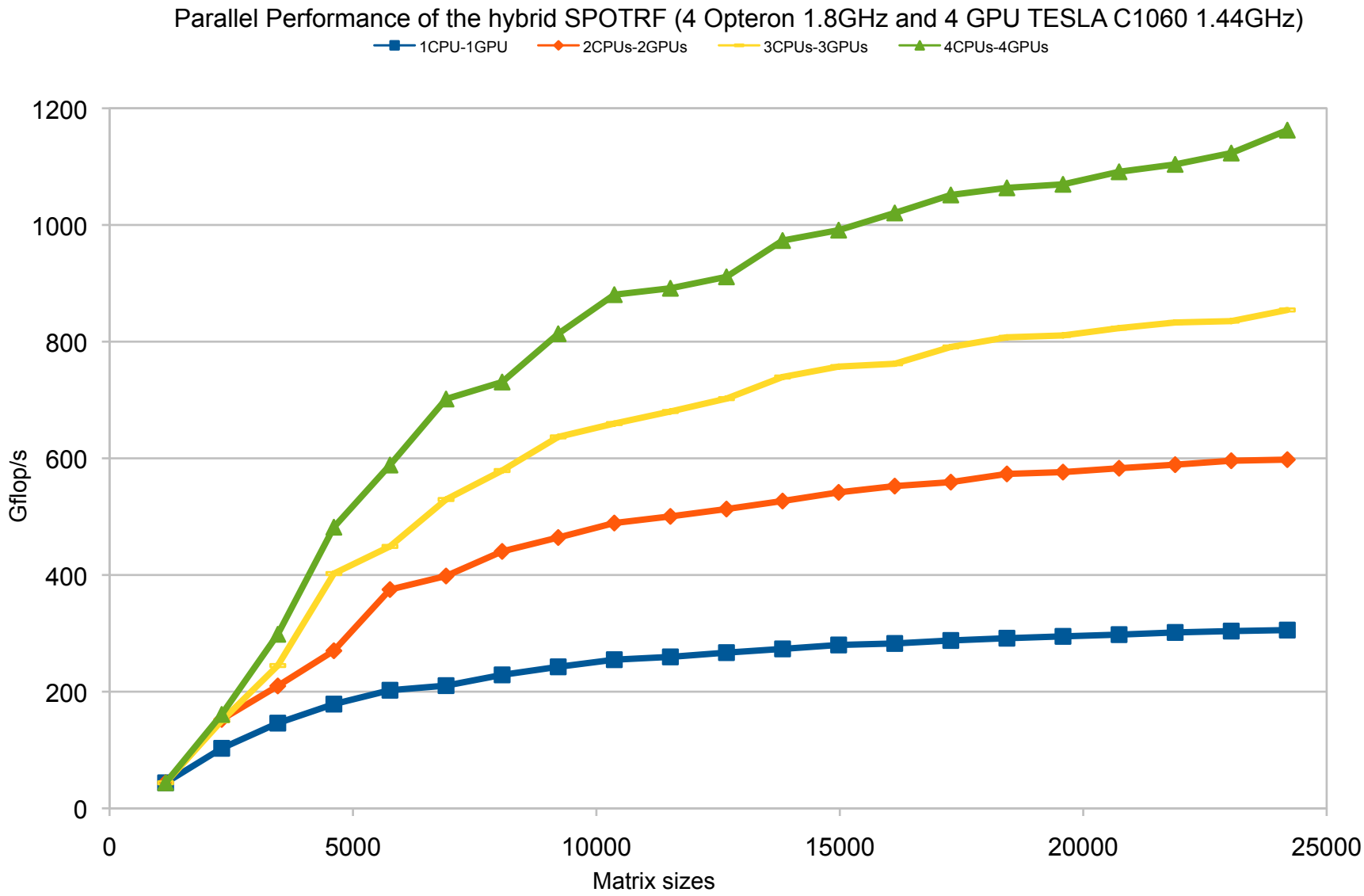


Current hybrid CPU+GPU algorithms
(small tasks for multicores and large tasks for GPUs)



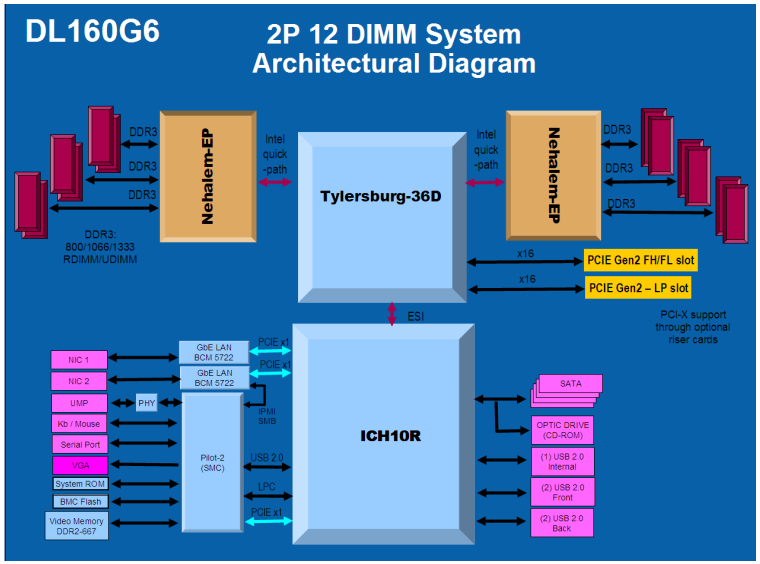
- **e.g. split the computation into tasks; define critical path that "clears" the way for other large data parallel tasks; proper schedule the tasks execution**
- **Design algorithms with well defined "*search space*" to facilitate auto-tuning**

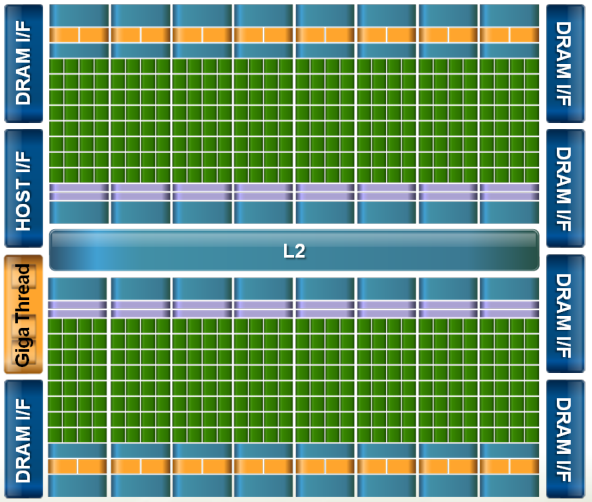
SP Cholesky on Multicore + Multi GPUs



NSF GPU Based System Keeneland

- Hewlett Packard Nodes
  - Dual socket Intel 2.8 GHz Nehalem-EP
  - 24 GB Main memory per node
- NVIDIA Servers
  - Fermi GPUs
- InfiniBand 4x QDR w/ full bisection interconnect
- Traditional Linux software stack augmented with GPU compilers, software tools, libraries
- Size: ~250 CPU sockets + ~250 GPU sockets
- Delivery and acceptance in Spring 2011





Georgia Tech NICS The University of Tennessee Oak Ridge National Laboratory NVIDIA hp NSF

A Call to Action

- Hardware has changed dramatically while software ecosystem has remained stagnant
- Need to exploit new hardware trends (e.g., manycore, heterogeneity) that cannot be handled by existing software stack, memory per socket trends
- Emerging software technologies exist, but have not been fully integrated with system software, e.g., UPC, Cilk, CUDA, HPCS
- Community codes unprepared for sea change in architectures
- No global evaluation of key missing components

International Exascale Software Program

Improve the world's simulation and modeling capability by improving the coordination and development of the HPC software environment

Workshops:

Build an international plan for coordinating research for the next generation open source software for scientific high-performance computing

www.exascale.org

International Community Effort

- **We believe this needs to be an international collaboration for various reasons including:**
  - The scale of investment
  - The need for international input on requirements
  - US, Europeans, Asians, and others are working on their own software that should be part of a larger vision for HPC.
  - No global evaluation of key missing components
  - Hardware features are uncoordinated with software development

Where We Are Today:

- SC08 (Austin TX) meeting to generate interest — Nov 2008
- Funding from DOE's Office of Science & NSF Office of Cyberinfratructure and sponsorship by Europeans and Asians
- US meeting (Santa Fe, NM) April 6-8, 2009 — Apr 2009
  - 65 people
- European meeting (Paris, France) June 28-29, 2009 — Jun 2009
  - Outline Report
- Asian meeting (Tsukuba Japan) October 18-20, 2009 — Oct 2009
  - Draft roadmap
  - Refine Report
- SC09 (Portland OR) BOF to inform others — Nov 2009
  - Public Comment; Draft Report presented
- European meeting (Oxford, UK) April 13-14, 2010 — Apr 2010
  - Refine and prioritize roadmap
  - Explore governance structure and management models
- Maui Meeting October 18-19, 2010 — Oct 2010
- Kobe Meeting - Spring 2011

Roadmap Purpose

- The IESP software roadmap is a planning instrument designed to enable the international HPC community to improve, coordinate and leverage their collective investments and development efforts.

- After we determine what needs to be accomplished, our task will be to construct the organizational structures suitable to accomplish the work

Roadmap Components

www.exascale.org

European Exascale Software Initiative - EESI

- A detailed evaluation of how Europe is positioned, its strengths and weaknesses, in the overall international HPC landscape and competition
  - Are European stakeholders willing/able to build an exa-scale prototype/by when?
  - Actors/users/projects
- A European and international vision and roadmap
  - Why is exa-scale initiatives important? Who cares? Impact?
    - Scientific
    - Economic
    - Social benefits
- Dissemination actions
  - Visibility of EESI: who is the potential target public?
    - R&D stakeholders
    - EC and national policy-makers
    - Society as a whole
- Identification of opportunities of worldwide collaborations
  - European position inside IESP: who's doing/deciding what?
  - Contribution to the international dialogs: mutual benefits!

EC and G8 Related Exascale Efforts

- G8 has a call out for "Interdisciplinary Program on Application Software towards Exascale Computing for Global Scale Issues"
  - 10 million € over three years
  - An initiative between Research Councils from Canada, France, Germany, Japan, Russia, the UK, and the USA
  - 78 preproposals submitted, 25 selected, expect to fund 6-10
  - Full proposals due August 25th
- EC FP7: Exascale computing, software and simulation
  - Announcement due September 28, 2010
  - 25 million €
  - 2 or 3 integrated project to be funded

Summary

- Major Challenges are ahead for extreme computing
  - Parallelism
  - Hybrid
  - Fault Tolerance
  - Power
  - … and many others not discussed here
- We will need completely new approaches and technologies to reach the Exascale level
- This opens up many new opportunities for computer science and applied mathematic research

If you are wondering what's beyond ExaFlops

Mega, Giga, Tera, Peta, Exa, Zetta …

10^{3}	kilo
10^{6}	mega
10^{9}	giga
10^{12}	tera
10^{15}	peta
10^{18}	exa
10^{21}	zetta

10^{24}	yotta
10^{27}	xona
10^{30}	weka
10^{33}	vunda
10^{36}	uda
10^{39}	treda
10^{42}	sorta
10^{45}	rinta
10^{48}	quexa
10^{51}	pepta
10^{54}	ocha
10^{57}	nena
10^{60}	minga
10^{63}	luma

- www.exascale.org

Jack Dongarra, Pete Beckman, Terry Moore, Jean-Claude Andre, Jean-Yves Berthou, Taisuke Boku, Franck Cappello, Barbara Chapman, Xuebin Chi, Alok Choudhary, Sudip Dosanjh, Al Geist, Bill Gropp, Robert Harrison, Mark Hereld, Michael Heroux, Adolfy Hoisie, Koh Hotta, Yutaka Ishikawa, Fred Johnson, Sanjay Kale, Richard Kenway, Bill Kramer, Jesus Labarta, Bob Lucas, Barney Maccabe, Satoshi Matsuoka, Paul Messina, Bernd Mohr, Matthias Mueller, Wolfgang Nagel, Hiroshi Nakashima, Michael E. Papka, Dan Reed, Mitsuhisa Sato, Ed Seidel, John Shalf, David Skinner, Thomas Sterling, Rick Stevens, William Tang, John Taylor, Rajeev Thakur, Anne Trefethen, Marc Snir, Aad van der Steen, Fred Streitz, Bob Sugar, Shinji Sumimoto, Jeffrey Vetter, Robert Wisniewski, Kathy Yelick

SPONSORS