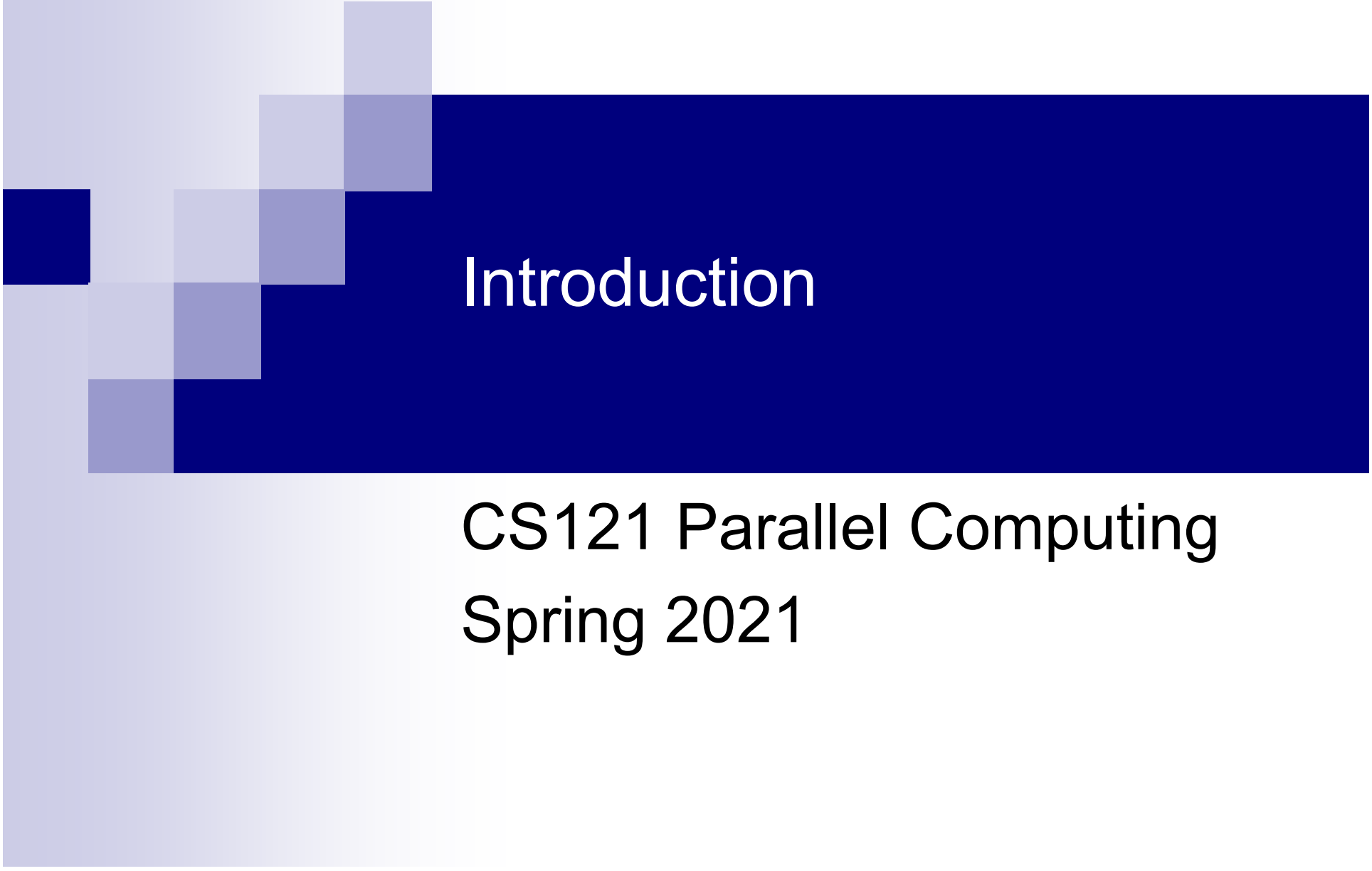


CS121 Parallel Computing

Spring 2021

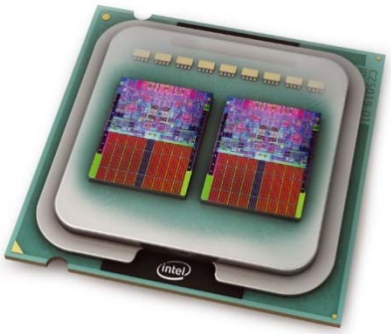
Course info

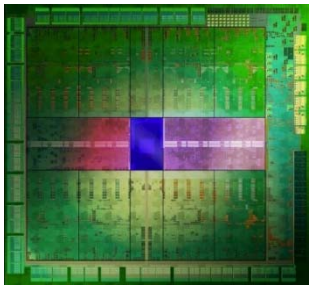
- Instructor: Assoc Prof Rui FAN 范睿
- Research: Parallel and distributed computing
- Contact: fanrui@shanghaitech.edu.cn (English please)
  20685372
- Office hours: Thursdays 5-6pm, SIST 1A-504E
- Recitation: TBA
- Website: Blackboard and Piazza

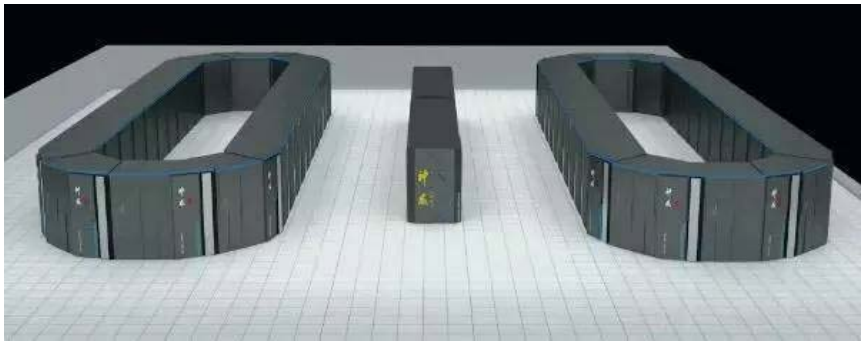
Problem sets	20%	▪ About once every 2 weeks
Labs	20%	▪ Solve problems using OpenMP and CUDA
Reading project	15% Teams of 2	▪ Find an interesting research paper from the suggested reading list ▪ Tell me your paper by week 8 ▪ Submit a report and give a 15 minute presentation in week 16
Course project	15% Teams of 2	▪ Find an interesting problem and write an efficient parallel program for it ▪ Tell me your problem by week 8 ▪ Submit a report and give a 15 minute presentation in week 16
Midterm exam	10%	▪ At start of week 9
Final exam	20%	
Attendance		

Parallel computing: what and why

- Parallel computing studies how to use multiple computers together to solve a problem.
- Allows solving complicated problems faster.
  - Ideally, with *k* processors we can solve a problem *k* times faster.
  - Also more memory to solve larger problems, or same problem with more accuracy.
  - May be more fault tolerant; but also more prone to faults.
- Almost all modern computer systems are parallel.
  - Multicores, GPUs, cloud computing, etc.
- Parallel computing crucial for modern large scale applications, e.g. physical simulations, data minining, machine learning.





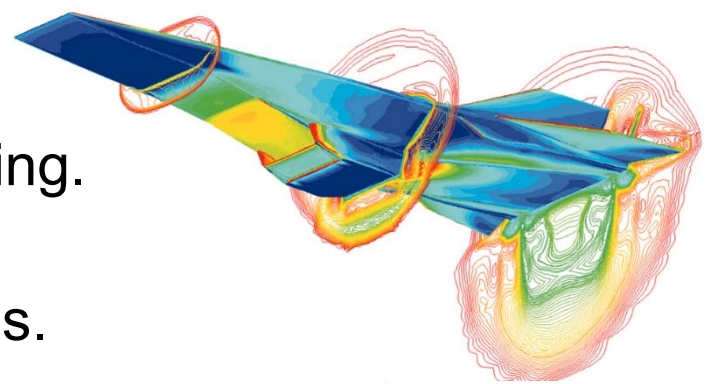


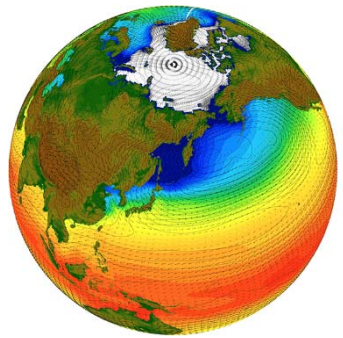
Course objectives

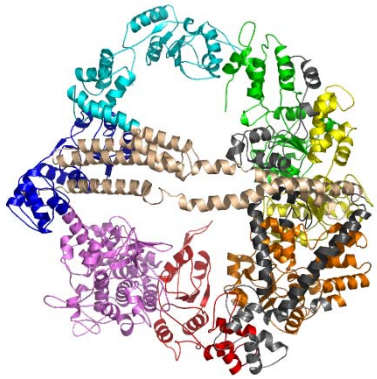
- To understand the concepts and techniques of parallel computing, and take advantage of the capabilities of modern systems.
  - Parallel hardware models and interaction with parallel software.
  - Power and limitations of parallelism.
  - Efficient parallel algorithms for important problems.

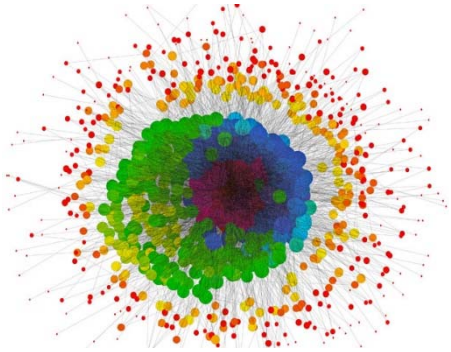
Applications

- Fluid dynamics, weather prediction, climate modeling.
- DNA, protein, drug structures and interactions.
- Quantum / atomic simulations, cosmological simulations.
- Cryptoanalysis.
- Big data analytics.
- Simulating financial and social behaviors.
- Machine learning and AI.
- Simulating the human brain.



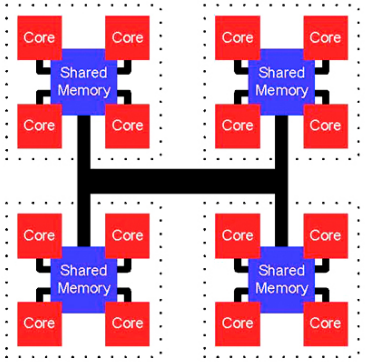


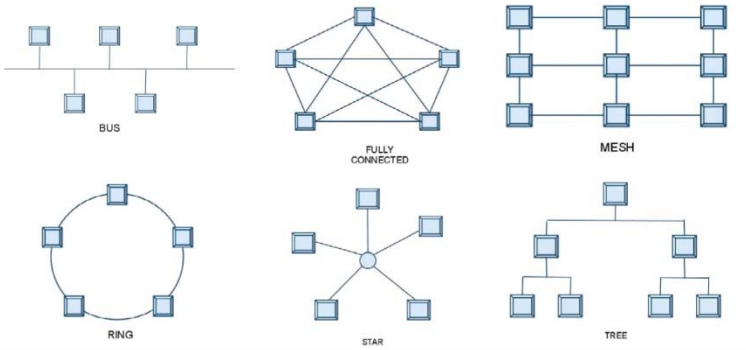


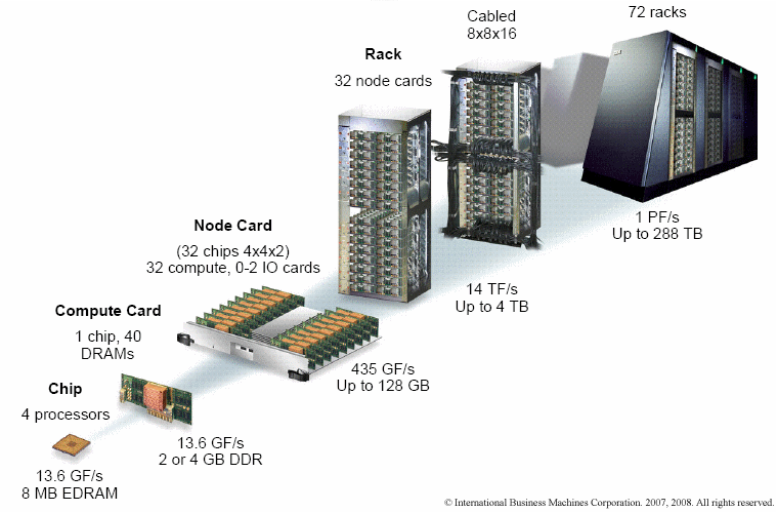


Parallel hardware

- Efficient parallel computing requires synergy between parallel hardware and software.
- Parallel system consists of multiple independent processors communicating over an interconnect.
- Unlike sequential (von Neumann) architecture, many parallel hardware designs.
  - Different types of processors (multicores, manycores, FPGA, etc.).
  - Heterogeneous designs combine multiple architectures, e.g. multicores and GPUs.
  - Different interconnect designs.
  - Communicate through shared memory, or message passing over network.
- Parallelism exists at many layers.
  - Instruction, core, chip, node, rack, etc.

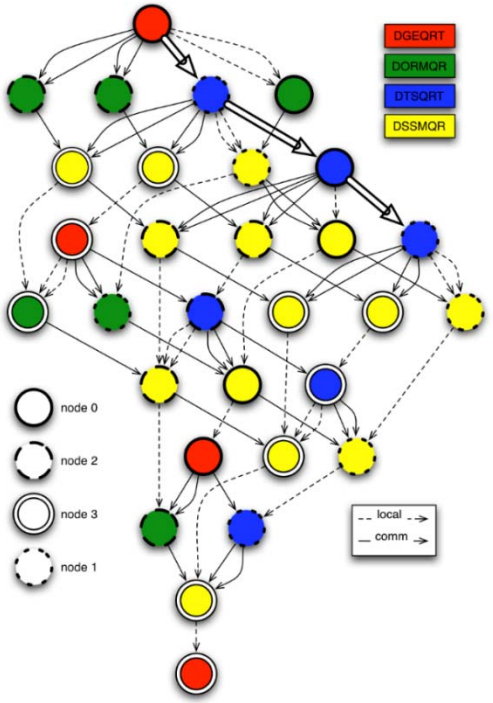






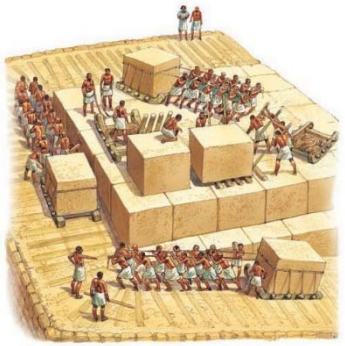
Parallel software

- Break a large problem into subproblems (tasks) that can be solved (somewhat) independently.
- OS and scheduler allocate tasks to different processors.
  - Respect dependencies between tasks.
- Parallel software must be matched to the hardware.
  - Similar amounts of concurrency in software and hardware.
  - Hardware must adequately handle software communication pattern.
  - No single hardware model suffices.
  - Parallel software is often not portable.
- PRAM model tries to abstract parallel hardware.
  - Useful for understanding inherent parallelism.
  - Unrealistically discounts cost of communication.



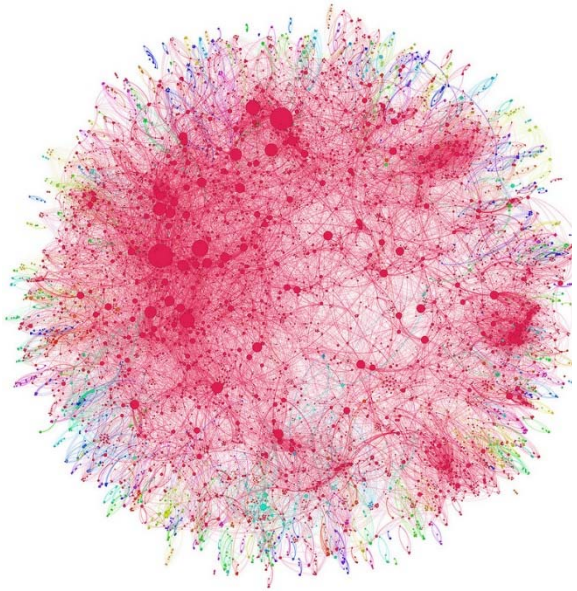
Challenges

- Harnessing power of the masses.
  - Easier said than done...
- Communication
  - Processors compute faster than they can communicate.
  - Problem gets worse as number of processors increases.
  - Main bottleneck to parallel computing.
- Synchronization
  - Tasks may interfere with each other, so can't be done at same time.
- Scheduling
  - Track and enforce dependencies.
  - Find good allocation of tasks to processors.
    - Data locality, heterogeneous processors
  - Maximize utilization and performance.



Challenges

- Structured vs unstructured
  - Structured problems can be solved with custom hardware.
  - Unstructured problems more general, but less efficient.
- Inherent limitations
  - Some problems are not (or don't seem to be) parallelizable.
    - Ex Binary search, Dijkstra's shortest paths algorithm.
  - Other problems require clever algorithms to become parallel.
    - Ex Fibonacci series ($a_n = a_{n-1} + a_{n-2}$).
- The human factor
  - Hard to keep track of concurrent events and dependencies.
  - Parallel algorithms are hard(er) to design and debug.



Course outline

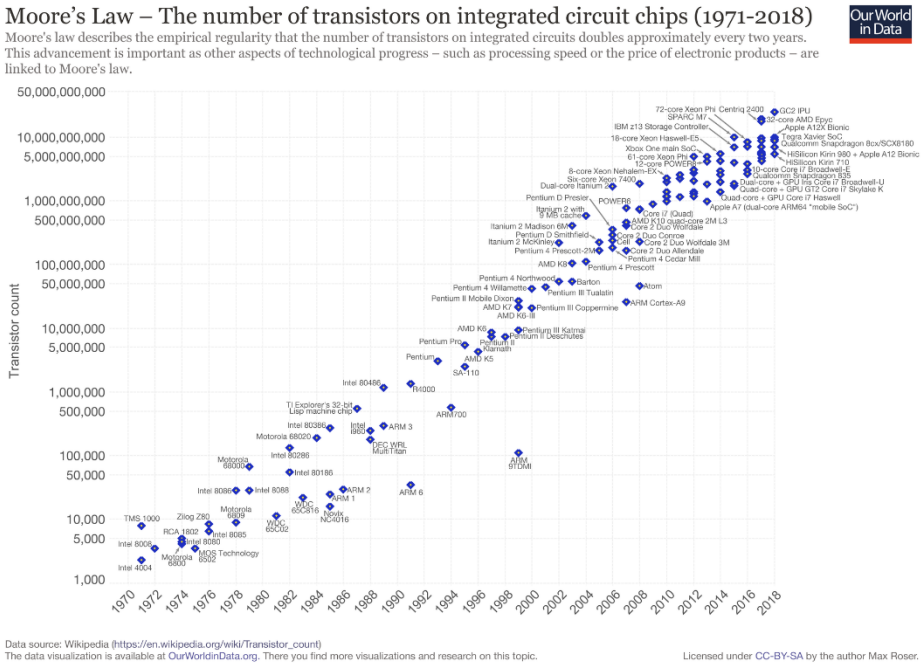
- Parallel architectures
  - Shared memory
  - Distributed memory
  - Manycore
- Parallel languages
  - OpenMP, MPI, CUDA, MapReduce
- Algorithm design techniques
  - Decomposition, load balancing, scheduling
- Parallel algorithms
  - Dense and sparse matrix algorithms, sorting, search, graph algorithms, PRAM algorithms, etc.

A brief history

- Research and theory started in the early 60's.
  - Cray-1 reached 160 MFLOPS in 1976.
- Commercially successful supercomputers (Cray, Thinking Machines, etc.) started in 1980's.
  - Used expensive custom processors.
- In 1990's massively parallel processors (MPPs) and clusters became dominant.
  - MPPs use commercial (OTS) processors with custom interconnects.
  - Clusters use OTS processors and interconnects running Linux.
    - Cheap, easy to build and relatively powerful.
    - Most data centers today are clusters.
- Fastest supercomputer today is Fujitsu Fugaku MPP.
  - Runs at 442 PFLOPS, about 3M times faster than a workstation.
- Apart from supercomputers, progress in parallel computing stalled in 1990's until mid 2000's.

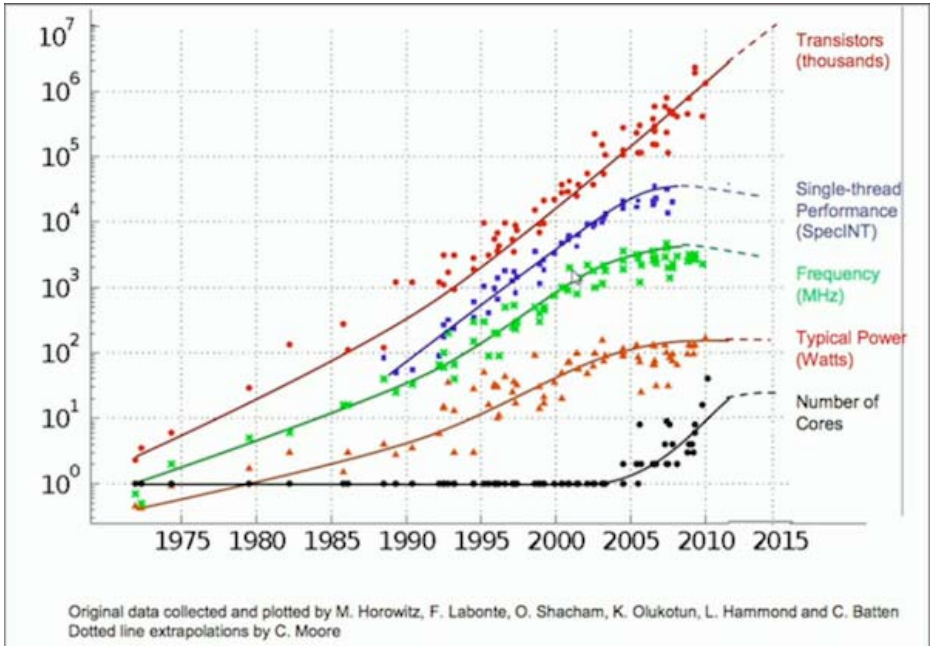
Moore's Law and parallel computing

- In 1965, Gordon Moore, co-founder of Intel, predicted transistor count would double every 18 months.
  - Held true for the last 50 years!
- Until mid 2000's, this implied single processor performance doubled at same rate.
- This held back development of parallel computers, since in the time to develop one, single processor performance would improve dramatically.
- But since ca. 2005, parallel processing has become essential to taking advantage of Moore's Law.



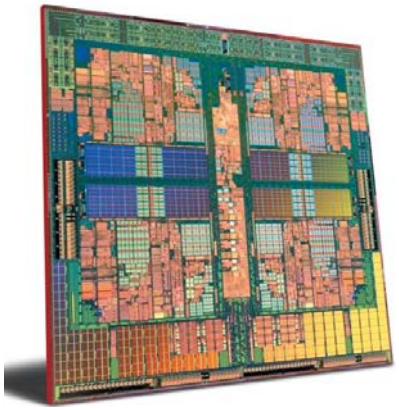
Moore's Law and performance

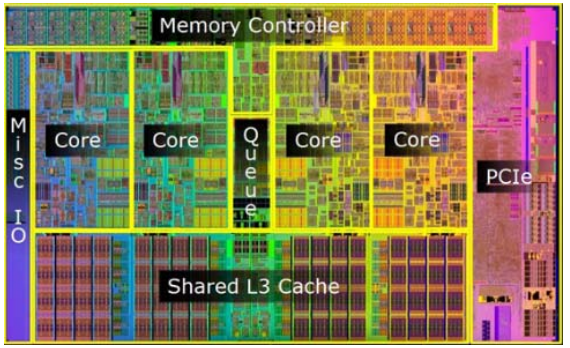
- Transistor properties, e.g. size and clock speed, do not scale equally.
- Higher single processor clock speeds is increasingly difficult to achieve.
  - Heat
  - Power consumption
  - Current leakage

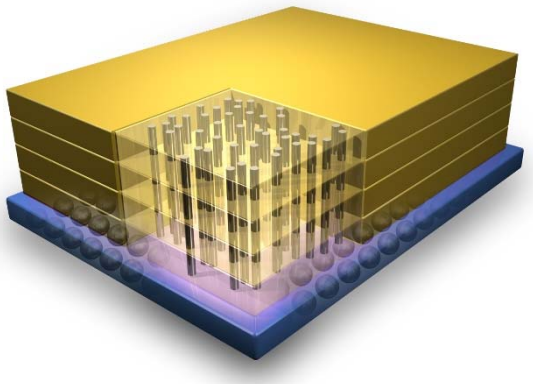


Moore's Law revisited

- Multicore technology addresses (lack of) clock speed scaling.
  - Link multiple processing cores together on same chip.
  - More efficient to replace a single high speed processor with multiple slower processors.
  - Another approach is to stack chips in a 3D structure.
- Developing software for multicores has been harder than scaling hardware.
  - Software developers with parallel computing skills are in high demand.







The state of the art

- Parallel computers today mainly based on four processor architectures.
  - Multicores
    - Small / moderate number (≤ 64) of fast, general purpose cores.
    - Ex Intel Xeon, IBM Power, Sun SPARC.
  - Manycores
    - Large number (1000's) of simple cores.
    - Ex Nvidia Pascal GPU, Intel Xeon Phi, Sunway SW26010.
  - FPGA (field programmable gate arrays)
    - Reconfigurable hardware customized for specific problems.
  - ASIC (application specific integrated circuits)
    - Specially built hardware for specific problems.
    - Ex Google TPU, Apple Neural Engine, IBM TrueNorth.
- In addition to processing speed, energy efficiency also increasing important.
  - Biggest datacenters consume over 100 MW of power, ~ 50K homes.
  - Biggest supercomputers consume ~ 20MW of power.
  - Goal is a supercomputer achieving 50 GFLOPS / W.
    - Current supercomputers achieve 1-6 GFLOPS / W.

Top 500 list

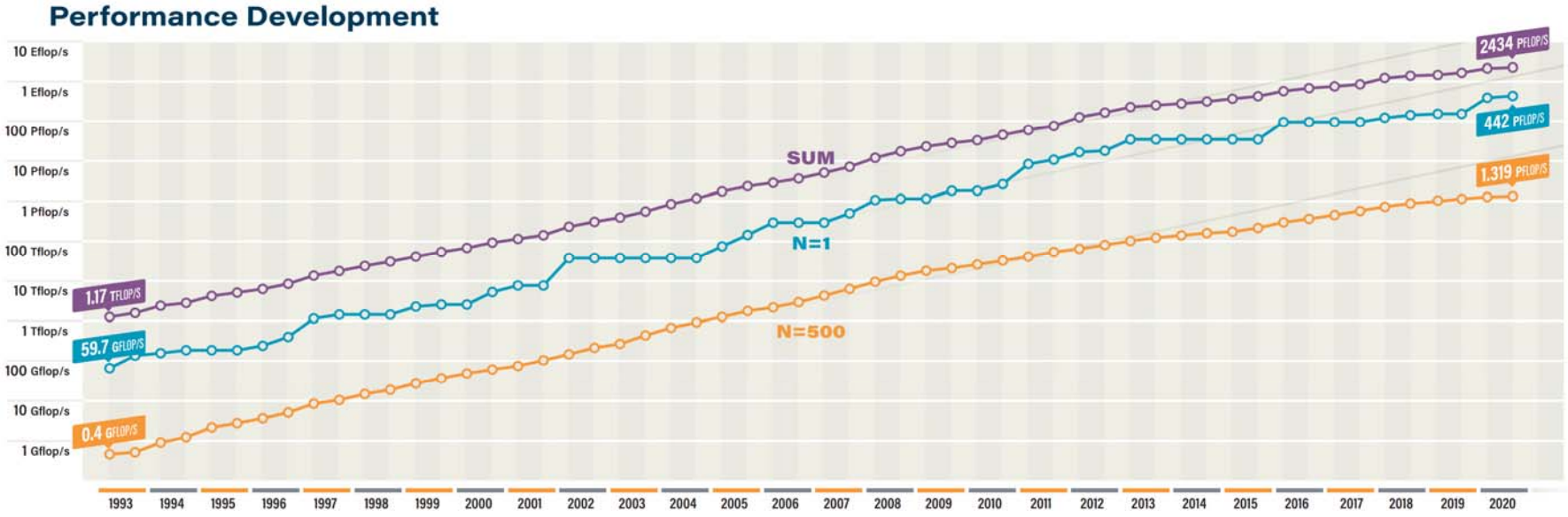
- Biannual ranking of fastest 500 supercomputers in the world.
  - Speed measured in floating point operations per second.
  - Uses high-performance LINPACK to solve a dense linear system Ax = b.
    - Compute intensive, but doesn't stress memory system.
    - May not represent performance on real-world problems.

NOVEMBER 2020	SYSTEM	SPECS	SITE	COUNTRY	CORES	RMAX PFLOP/S	POWER MW
1	Fugaku	Fujitsu A64FX (48C, 2.2GHz), Tofu Interconnect D	RIKEN R-CCS	Japan	7,630,848	442.0	29.9
2	Summit	IBM POWER9 (22C, 3.07GHz), NVIDIA Volta GV100 (80C), Dual-Rail Mellanox EDR Infiniband	DOE/SC/ORNL	USA	2,414,592	148.6	10.1
3	Sierra	IBM POWER9 (22C, 3.1GHz), NVIDIA Tesla V100 (80C), Dual-Rail Mellanox EDR Infiniband	DOE/NNSA/LLNL	USA	1,572,480	94.6	7.44
4	Sunway TaihuLight	Shenwei SW26010 (260C, 1.45 GHz) Custom Interconnect	NSCC in Wuxi	China	10,649,600	93.0	15.4
5	Selene	NVIDIA DGX A100, AMD EPYC 7742 (64C, 2.25GHz), NVIDIA A100, Mellanox HDR Infiniband	NVIDIA Corporation	USA	555,520	63.4	2.65

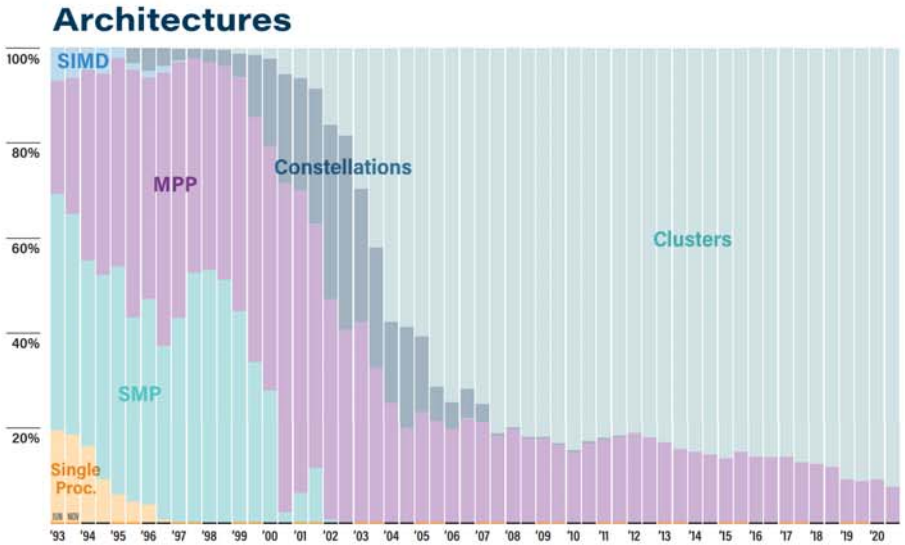
Mega	Giga	Tera	Peta	Exa
10^6	10^9	10^{12}	10^{15}	10^{18}

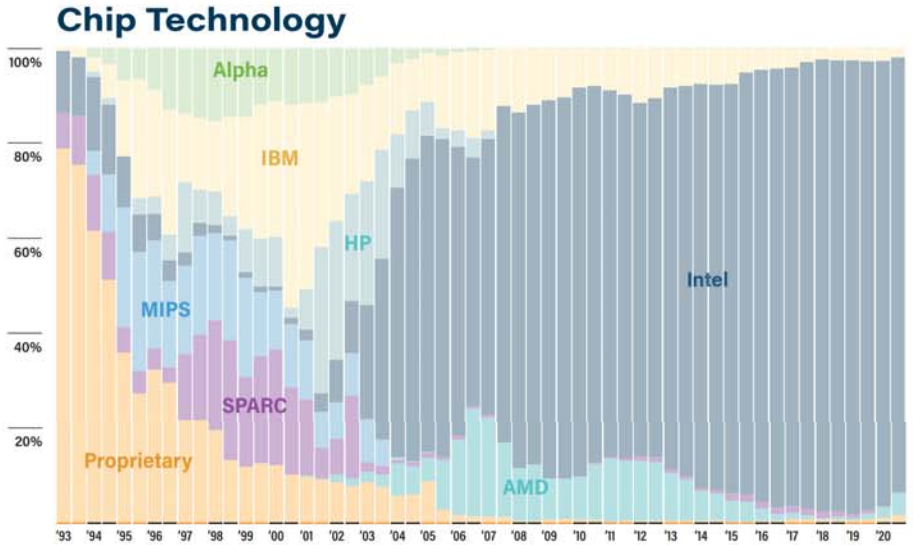
  - For comparison, Intel multicore achieves ~50 GFLOPS / core, and GPU achieves ~ 10 TFLOPS / board.

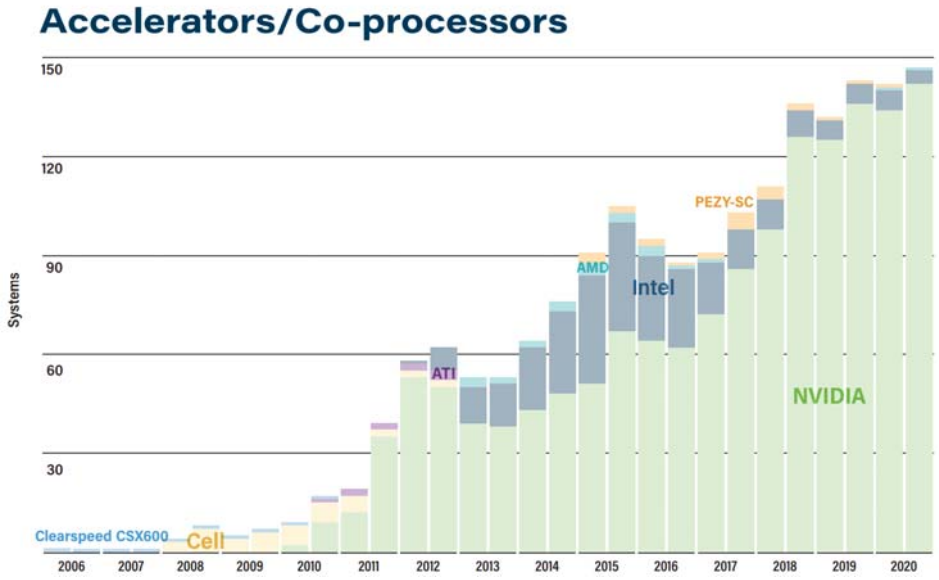
Top 500 – Trends



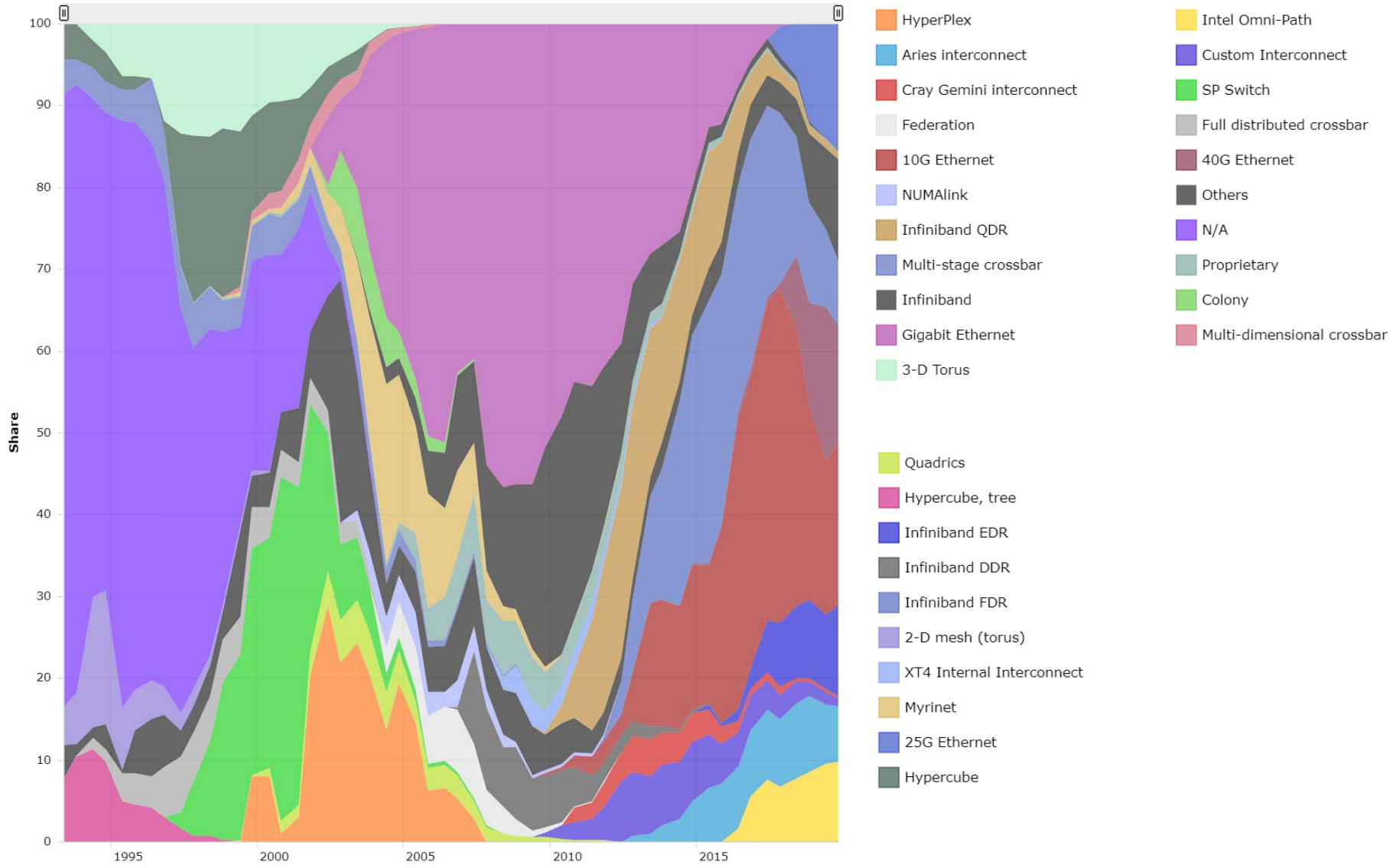
Top 500 – Architecture



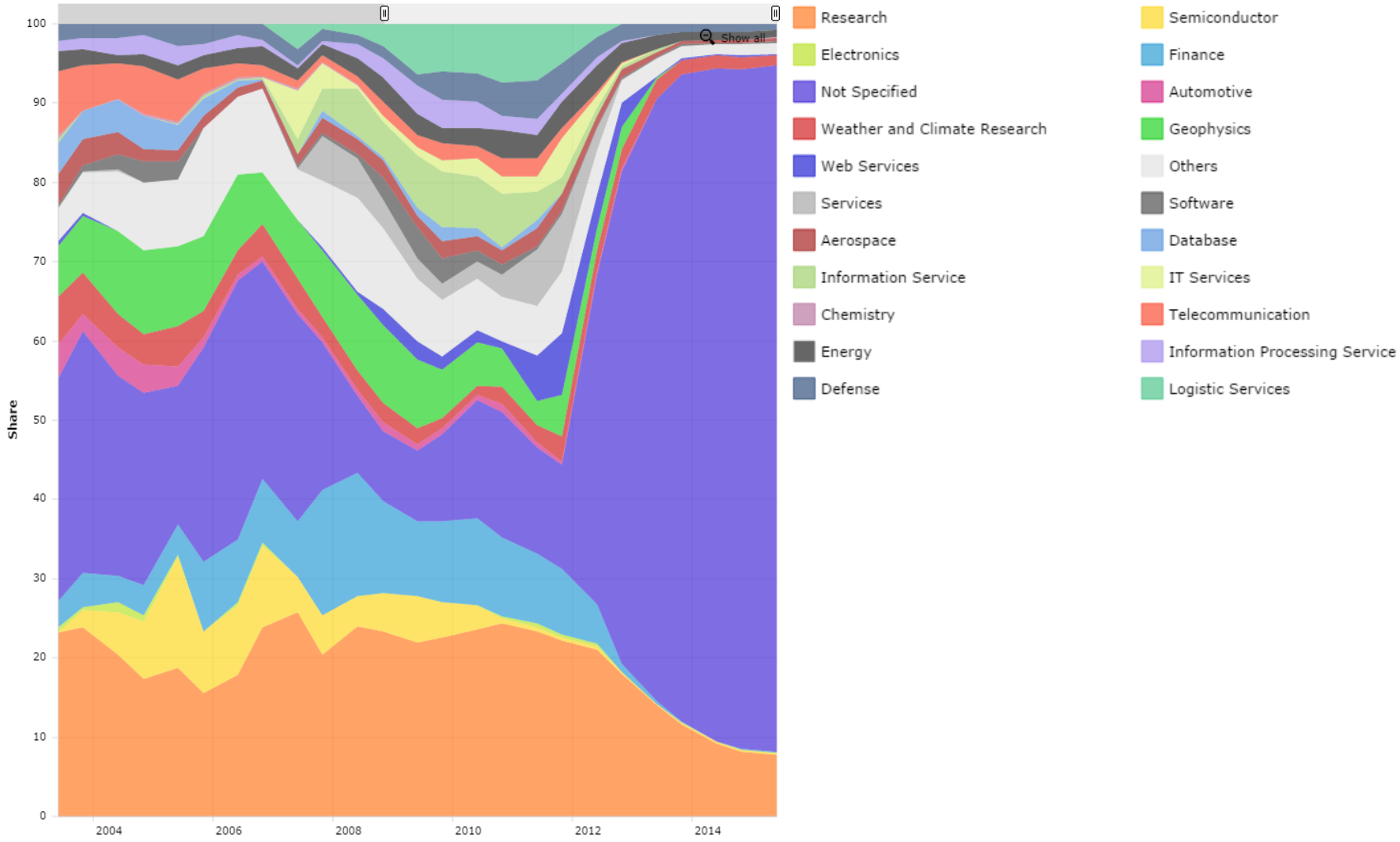




Top 500 – Interconnect



Top 500 – Applications



Other performance measures

- LINPACK does compute-intensive operations on structured dense matrices.
  - Uniform control flow, predictable and coalesced memory accesses.
  - Ideal for physical simulations.
- Data-intensive applications today have instruction divergence, branching and random memory accesses.
- New benchmarks give more complete performance picture
  - HPCG performs sparse matrix operations.
  - Graph500 performs breadth-first search.
- A computer's performance can differ dramatically depending on benchmark.

HPCG

New HPCG results announced at ISC 2020

Rank	Site	Computer	Cores	HPL Rmax (Pflop/s)	TOP500 Rank	HPCG (Pflop/s)	Fraction of Peak
1	RIKEN Center for Computational Science Japan	**Supercomputer Fugaku** – Fujitsu A64X, A64FX 6635520C 2.2GHz, Tofu Interconnect D Fujitsu	6,635,520	415.530	1	13.366	2.6%
2	DOE/SC/ORNL USA	**Summit** – AC922, IBM POWER9 22C 3.07GHz, dual-rail Mellanox EDR Infiniband, NVIDIA Volta V100 IBM	2,414,592	148.600	2	2.926	1.5%
3	DOE/NNSA/LLNL USA	**Sierra** – S922LC, Power9 22C 3.1GHz, Mellanox EDR, NVIDIA Tesla V100 IBM / NVIDIA / Mellanox	1,572,480	94.640	3	1.796	1.4%
4	Eni Green Data Center Italy	**HPC5** – PowerEdge C4140, Intel Xeon-Gold 6252 24C 2.1GHz, Infiniband HDR100, Nvidia Tesla V100 PCIe 32GB Dell EMC	128	35.450	6	0.860	1.7%
5	DOE/NNSA/LANL/SNL USA	**Trinity** – Cray XC40, Intel Xeon E5-2698 v3 16C 2.3GHz, Aries, Intel Xeon Phi 7250 68C 1.4GHz Cray	979,072	20.159	11	0.546	1.3%
6	NVIDIA USA	**Selene** – NVIDIA DGX A100, AMD EPYC 7742 35200C 2.25GHz, Mellanox HDR InfiniBand, NVIDIA Tesla A100 40GB NVIDIA	272,800	27.580	7	0.509	1.5%
7	National Institute of Advanced Industrial Science and Technology (AIST) Japan	**AI Bridging Cloud Infrastructure (ABCI)** – PRIMERGY CX2570M4, Intel Xeon Gold 6148 20C 2.4GHz, Infiniband EDR, NVIDIA Tesla V100 Fujitsu	391,680	19.880	12	0.509	1.6%
8	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** – Cray XC50, Intel Xeon E5-2690v3 12C 2.6GHz, Cray Aries, NVIDIA Tesla P100 16GB Cray	387,872	21.230	10	0.497	1.8%
9	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** – Sunway MPP, SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93.015	4	0.481	0.4%
10	Korea Institute of Science and Technology Information Republic of Korea	**Nurion** – CS500, Intel Xeon Phi 7250 68C 1.4GHz, Intel Omni-Path, Intel Xeon Phi 7250 Cray	570,020	13.929	18	0.391	1.5%

Graph500

RANK	PREVIOUS RANK	MACHINE	VENDOR	TYPE	NETWORK	INSTALLATION SITE	LOCATION	COUNTRY	YEAR	APPLICATION	USAGE	NUMBER OF NODES	NUMBER OF CORES	MEMORY	IMPLEMENTATION	SCALE	GTEPS
1	1	K computer	Fujitsu	Custom	Tofu	RIKEN Advanced Institute for Computational Science (AICS)	Kobe Hyogo	Japan	2011	Various scientific and instudrial fields	Academic and industry	82944	663552	1327100 gigabytes	Custom	40	31302.4
2	2	Sunway TaihuLight	NRCPC	Sunway MPP	Sunway	National Supercomputing Center in Wuxi	Wuxi	China	2015	research	research	40768	10599680	1304580 gigabytes	Custom	40	23755.7
3	3	DOE/NNSA/LLNL Sequoia	IBM	BlueGene/Q Power BQC 16C 1.60 GHz	Custom	Lawrence Livermore National Laboratory	Livermore CA	USA	2012	Scientific Research	Government	98304	1572864	1572860 gigabytes	Custom	41	23751
4	4	DOE/SC/Argonne National Laboratory Mira	IBM	BlueGene/Q Power BQC 16C 1.60 GHz	Custom	Argonne National Laboratory	Chicago IL	USA	2012	Scientific Research	Research	49152	786432	786432 gigabytes	Custom	40	14982
5	new	SuperMUC-NG	Lenovo	ThinkSystem SD530 Xeon Platinum 8174 24C 3.1GHz Intel Omni-Path		Leibniz Rechenzentrum	Garching	Germany	2018	Academic	Research	4096	196608	393216	custom-aml-heavy-dropt	39	6279.47
6	5	JUQUEEN	IBM	BlueGene/Q Power BQC 16C 1.60 GHz	Custom	Forschungszentrum Juelich (FZJ)	Juelich	Germany	2012	Scientific Research	Research	16384	262144	262144 gigabytes	Custom	38	5848
7	6	ALCF Mira - 8192 partition	IBM	IBM - BlueGene/Q Power BQC 16C 1.60 GHz		Argonne National Laboratory	Chicago IL	United States	2012	Scientific Research	Research	8192	131072	131072 gigabytes	Custom	36	4212
8	8	Fermi	IBM	BlueGene/Q Power BQC 16C 1.60 GHz	Custom	CINECA	Casalecchio Di Reno	Italy	2012	Scientific Research	Academic	8192	131072	131072 gigabytes	Custom	37	2567
9	new	NERSC Cori - 1024 haswell partition	Cray	XC40	Aries	NERSC/LBNL	DOE/SC/LBNL/NERSC	United States	2017	Government	basic science and simulation	1024	32768	133376	custom-aml-heavy-dropt	37	2562.16
10	9	ALCF Mira - 4096 partition	IBM	IBM - BlueGene/Q Power BQC 16C 1.60 GHz	5D Torus	Argonne National Laboratory	Chicago IL	United States	2012	Scientific Research	Research	4096	65536	65536 gigabytes	Custom	35	2348

HPCL State Key Laboratory of High Performance Computing

Overview of Tianhe-2 (MilkyWay-2) Supercomputer

Yutong Lu

School of Computer Science, National University of Defense Technology;
State Key Laboratory of High Performance Computing, China
ytlu@nudt.edu.cn

国防科学技术大学
National University of Defense Technology

Motivation

Tianhe-2 (Milkyway-2) Supercomputer

Specification

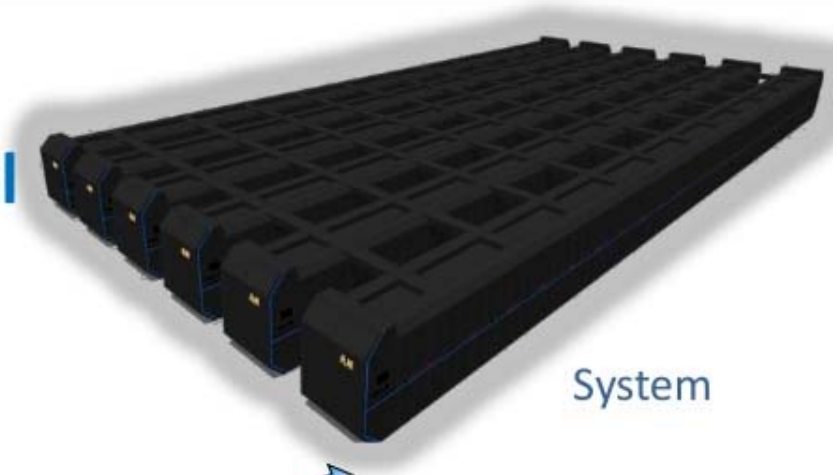
■ Hybrid Architecture

◆ Xeon CPU & Xeon Phi

Items	Configuration
Processors	32000 Intel Xeon CPUs + 48000 Xeon Phis + 4096 FT CPUs Peak performance is 54.9PFlops, HPL
Interconnect	Proprietary high-speed interconnection network TH Express-2
Memory	1.4PB in total
Storage	Global shared parallel storage system, 12.4PB
Cabinets	125+13+24=162 compute/communication/storage Cabinets
Power	17.8 MW (1902MFlops/W)
Cooling	Closed Air cooling system

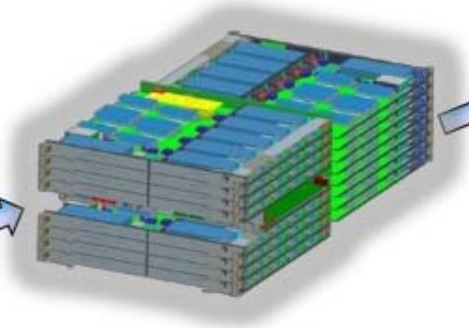
From Chips to Entire System

- 16000 compute nodes in total
- Frame: 32 compute Nodes
- Rack: 4 Compute Frames
- Whole System: 125 Racks



System

Compute Node



Compute Frame

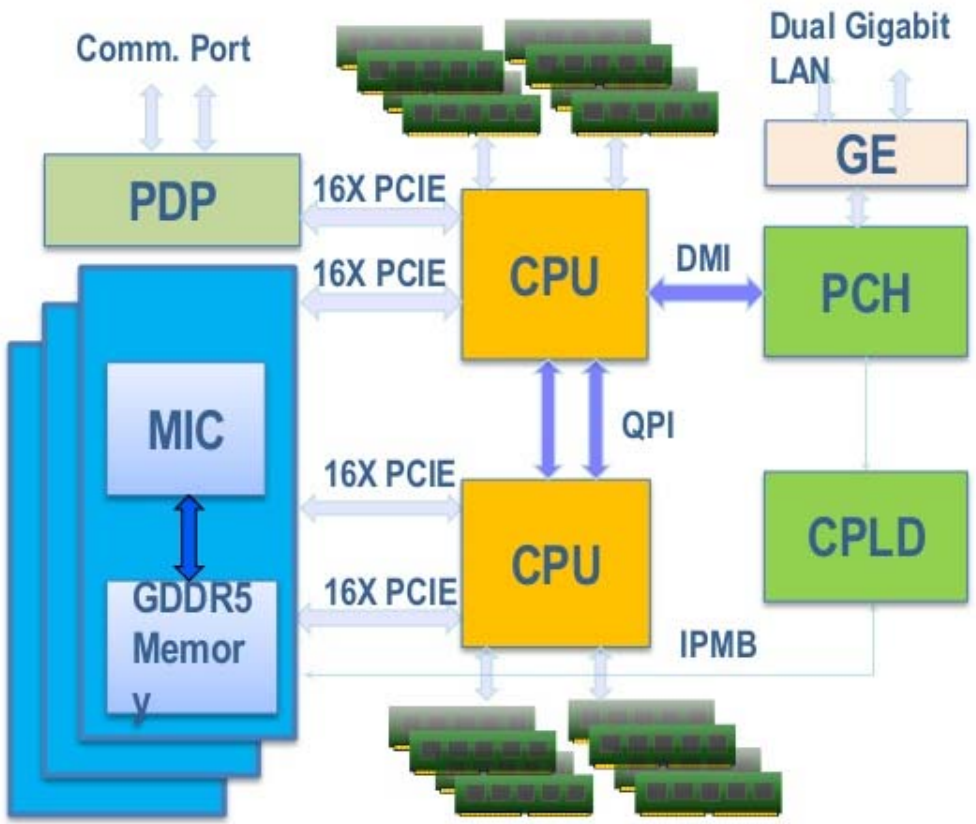
Compute Rack

国防科学技术大学
National University of Defense Technology

Compute Node

■ Neo-Heterogeneous Compute Node

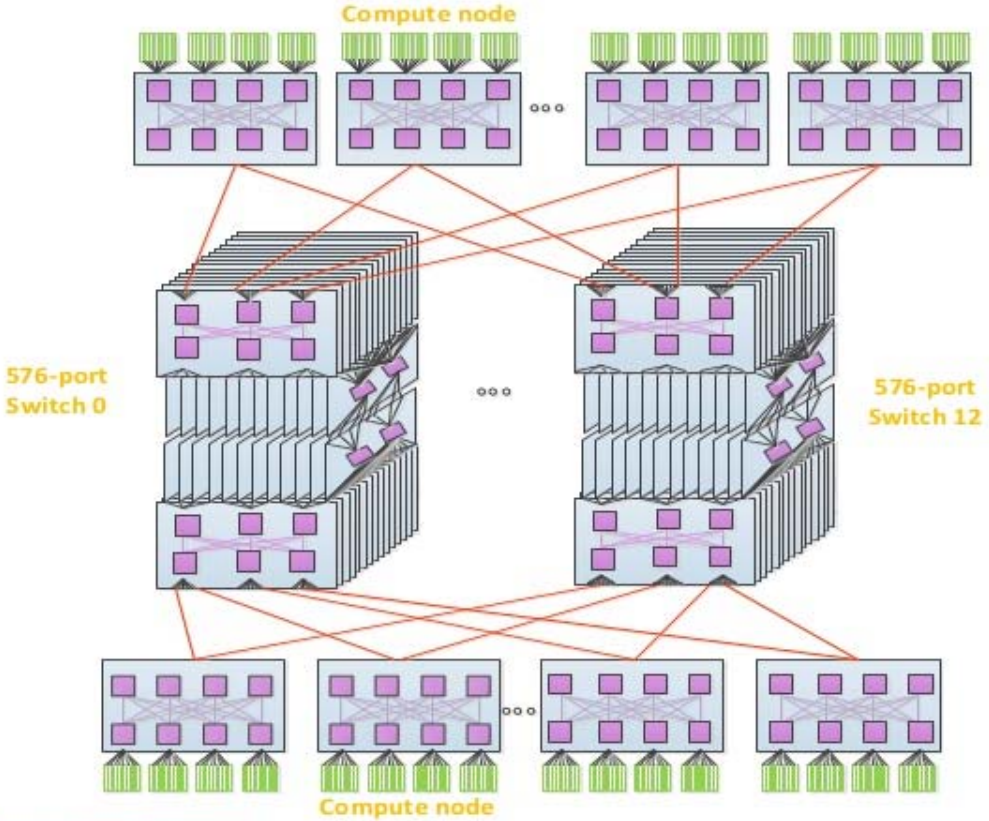
- ◆ Similar ISA, different ALU
- ◆ 2 Intel Ivy Bridge CPU + 3 Intel Xeon Phi
- ◆ 16 Registered ECC DDR3 DIMMs, 64GB
- ◆ 3 PCI-E 3.0 with 16 lanes
- ◆ PDP Comm. Port
- ◆ Dual Gigabit LAN
- ◆ Peak Perf. : 3.432Tflops



Interconnection network

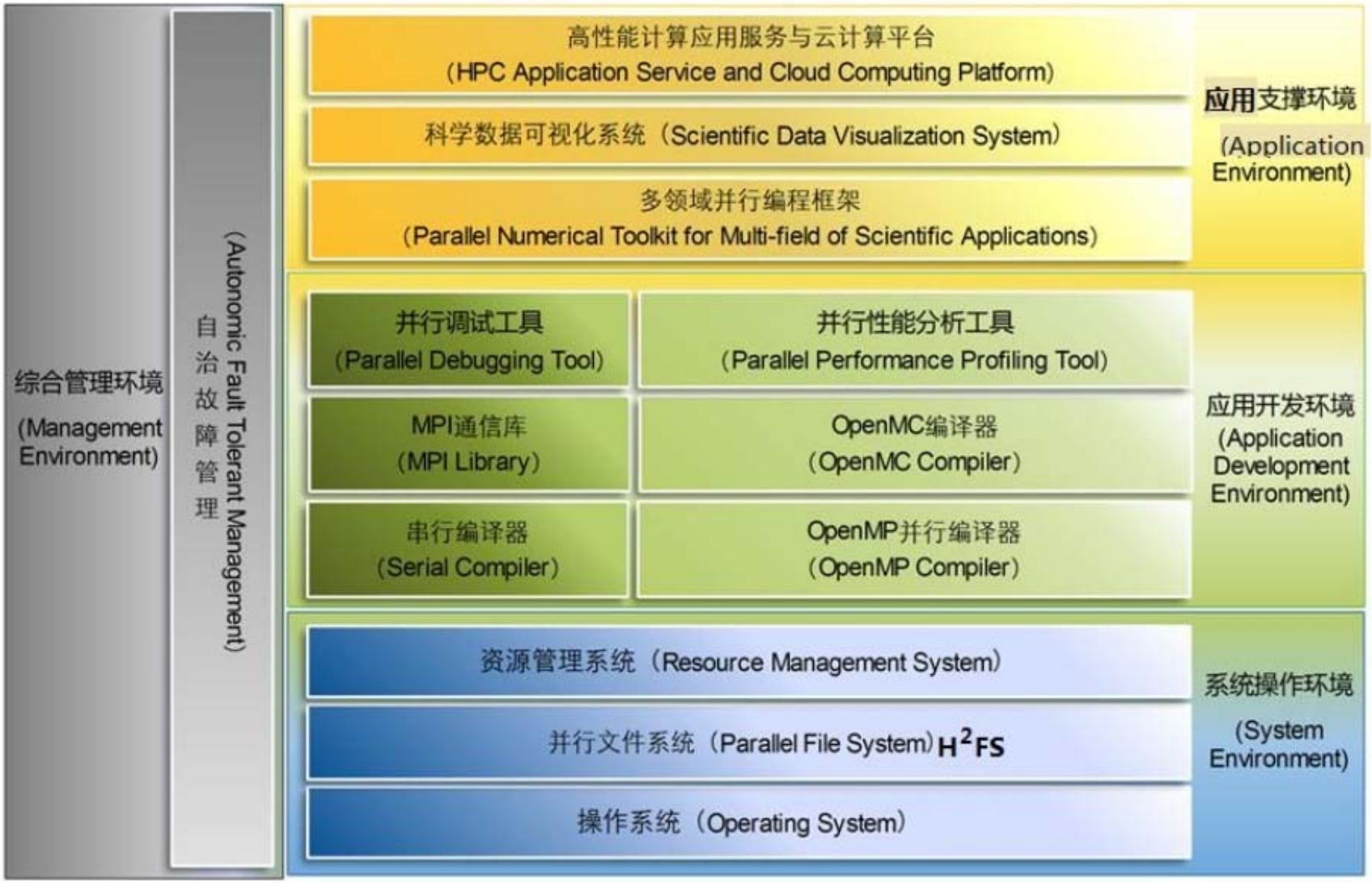
■ TH Express-2 interconnection network

- ◆Fat-tree topology using 13 576-port top level switches
- ◆Opto-electronic hybrid transport tech.
- ◆Proprietary network protocol
- ◆NRC +NIC



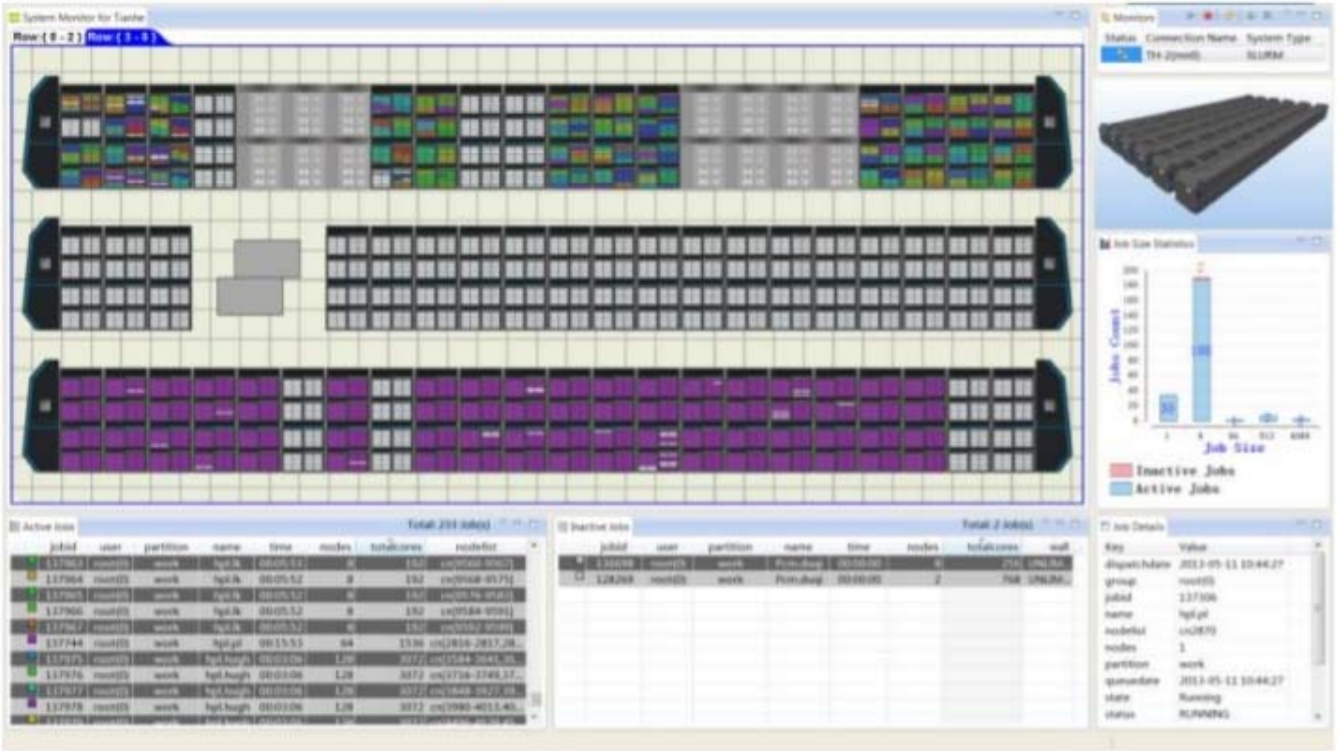
国防科学技术大学
National University of Defense Technology

HPC Software stack



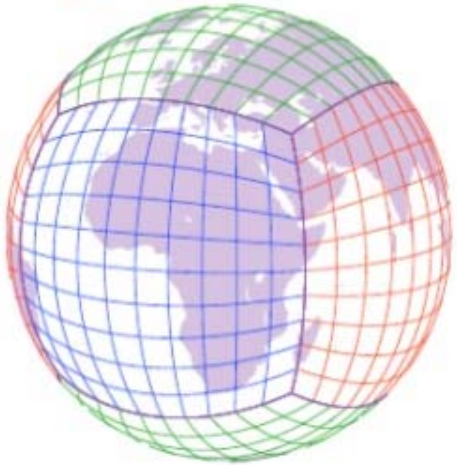
OS & RMS

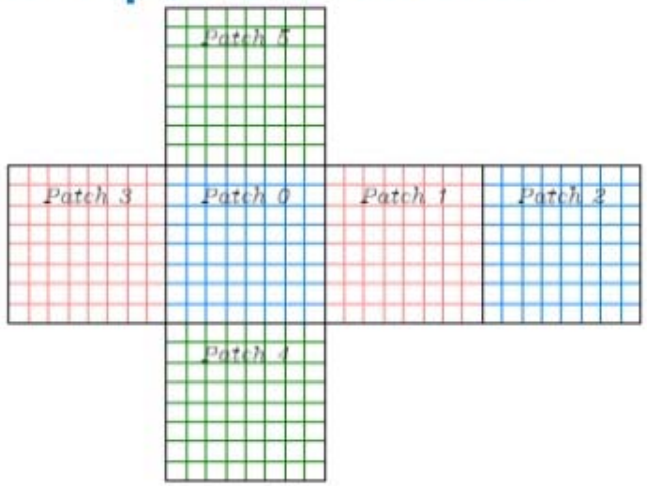
- **Operating System**
  - Kylin Linux
- **Resource manage system**
  - Power-aware resource allocation
  - Multiple custom schedule policies



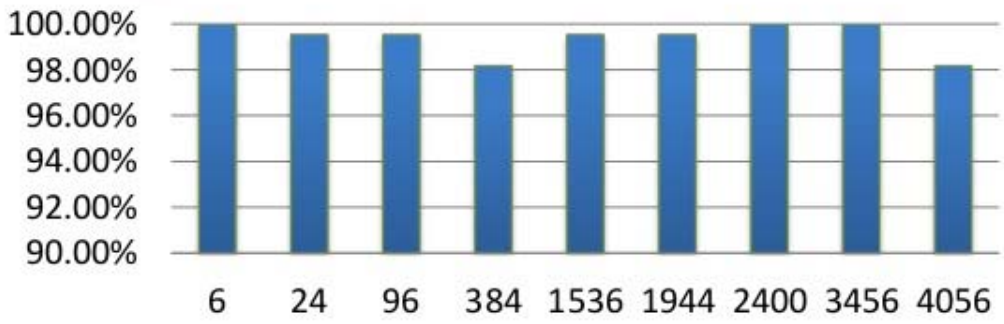
Application

- Application of a global shallow water model: algorithms
  - Hierarchical data partition & communication on cubed-sphere
  - Balanced partition between CPU/MIC inside each node
  - Communication hiding algorithm based on "Pipe-flow" scheme





- Nearly ideal weak scaling on the Tianhe-2
  - Using up to 4,056 nodes (97,344 CPU cores + 693,576 MIC cores)
  - # of unknowns for the largest run: 200 billion



Slides provided by Chao Yang from Institute of Software, Chinese Academy of Sciences

国防科学技术大学
National University of Defense Technology

Course texts

- Course materials partly taken from the following texts.
  - But all topics covered by lecture slides.
- *Introduction to Parallel Computing*. Grama, Karypis, Kumar, Gupta. Pearson, 2003.
- *An Introduction to Parallel Programming*. Peter Pacheco. Morgan Kaufmann 2011.
- *Programming Massively Parallel Processors*. Kirk, Hwu. Morgan Kaufmann 2016.
- *CUDA by Example*. Sanders, Kandrot. Addison-Wesley 2010.