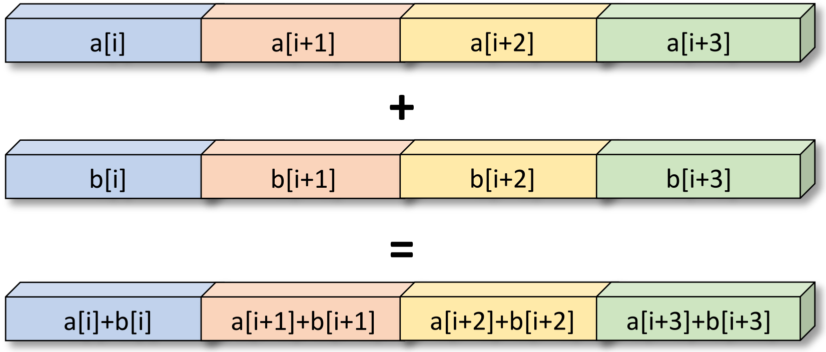
What Every Computational Researcher Should Know About Computer Architecture*



*Plus some other useful technology

Ian A. Cosden, Ph.D.
Manager, HPC Software Engineering and Performance Tuning
Research Computing, Princeton University

Princeton Research Computing Bootcamp
October 29, 2018

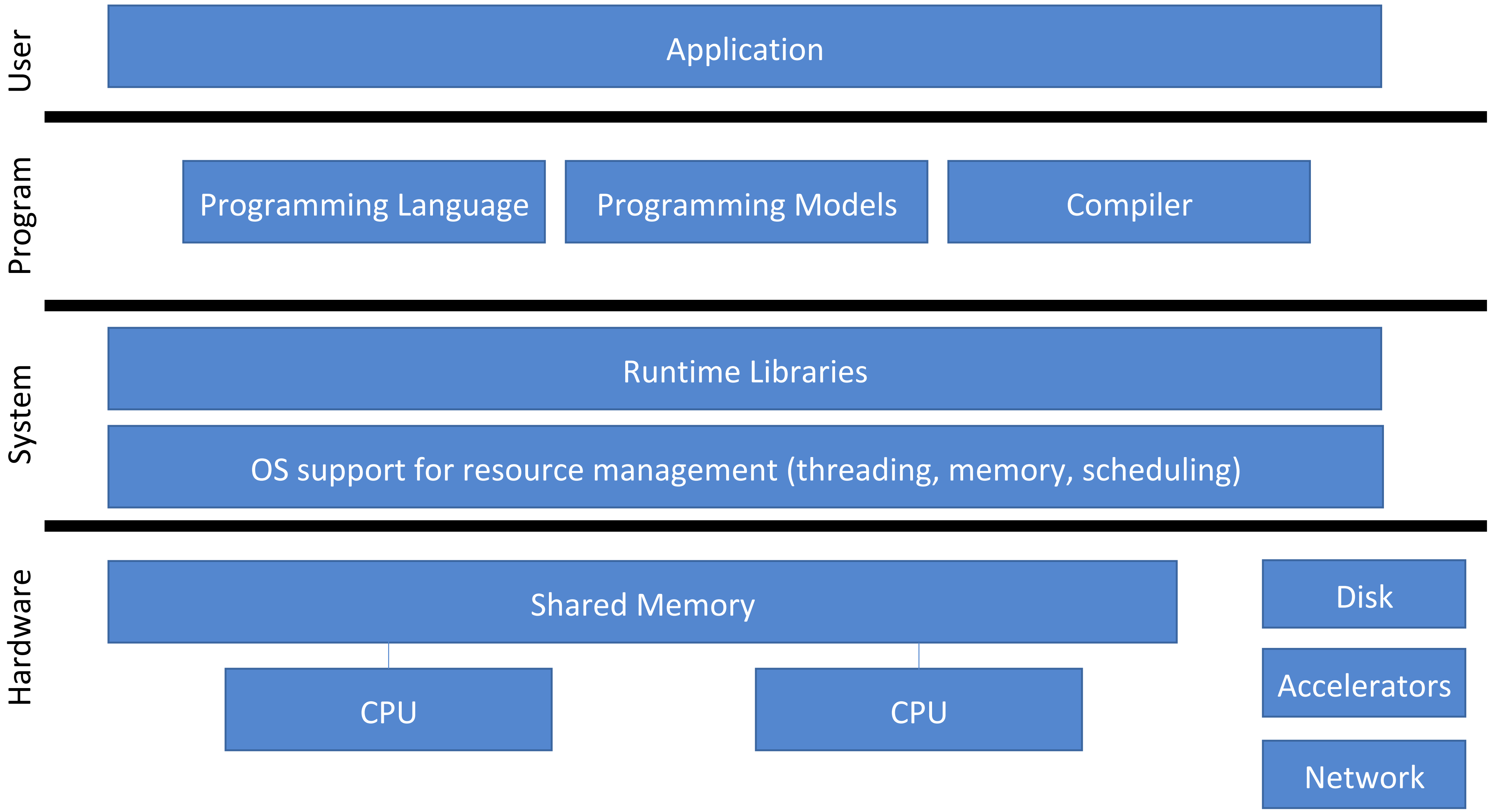
Outline

- Motivation
- HPC Cluster
- Compute
- Memory
- Disk
- Emerging Architectures

Why Care About the Hardware?

- Get the most of out the next few days
- Be able to ask better questions and understand the answers
- Prevent programming mistakes
- Make educated decisions
- Architecture changes can force new design and implementation strategies
- Reproducibility
- Exploit the hardware to your advantage (performance)

Software Stack



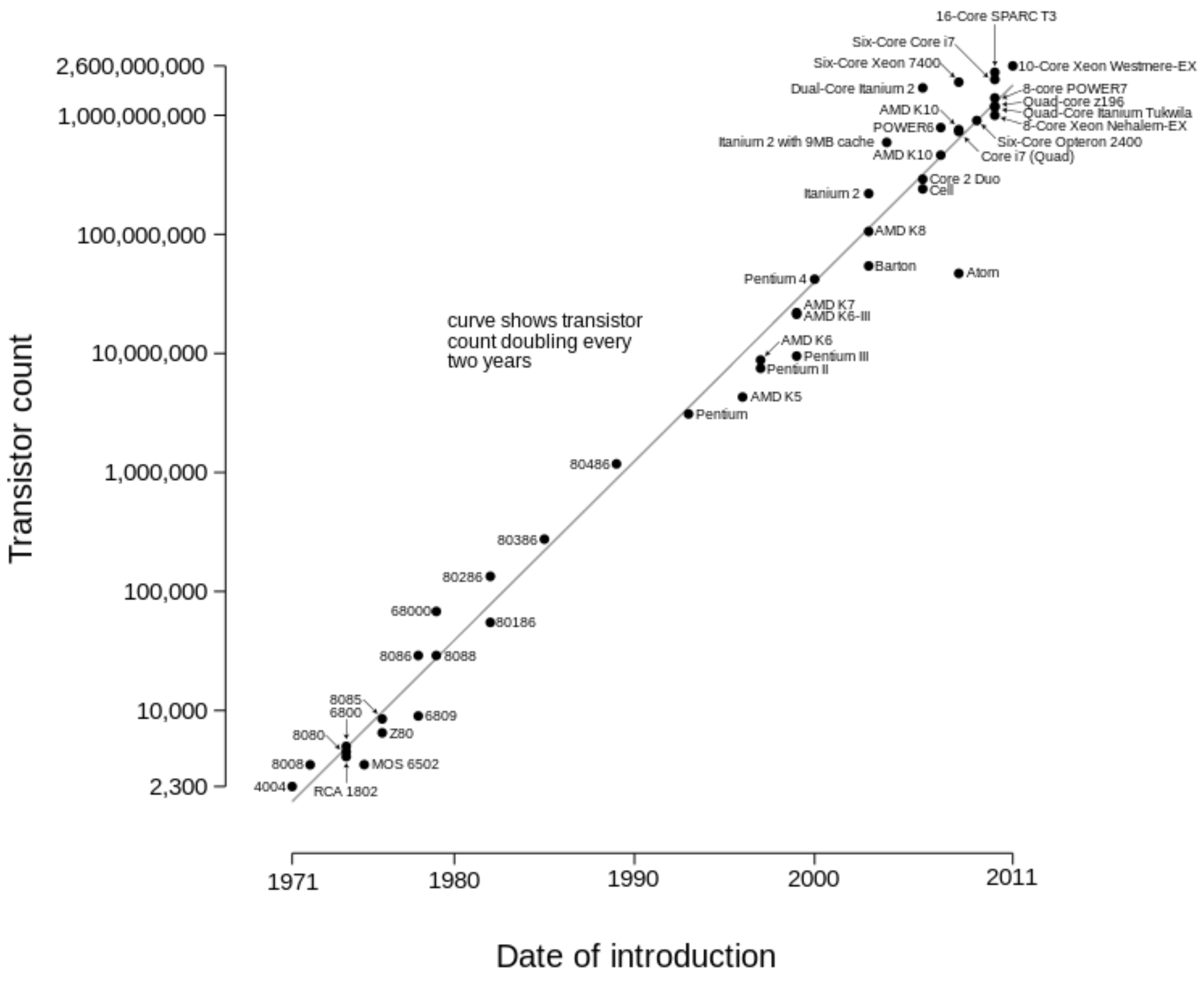
Why Care About HPC Hardware?

- High-Performance Computing (HPC)
  - Aggregate computing power in order to deliver far more capacity than a single computer
  - Supercomputers
- Some problems demand it from the onset
- Many problems evolve to need it
  - When you outgrow your laptop, desktop, departmental server
  - Need to know capabilities and limitations

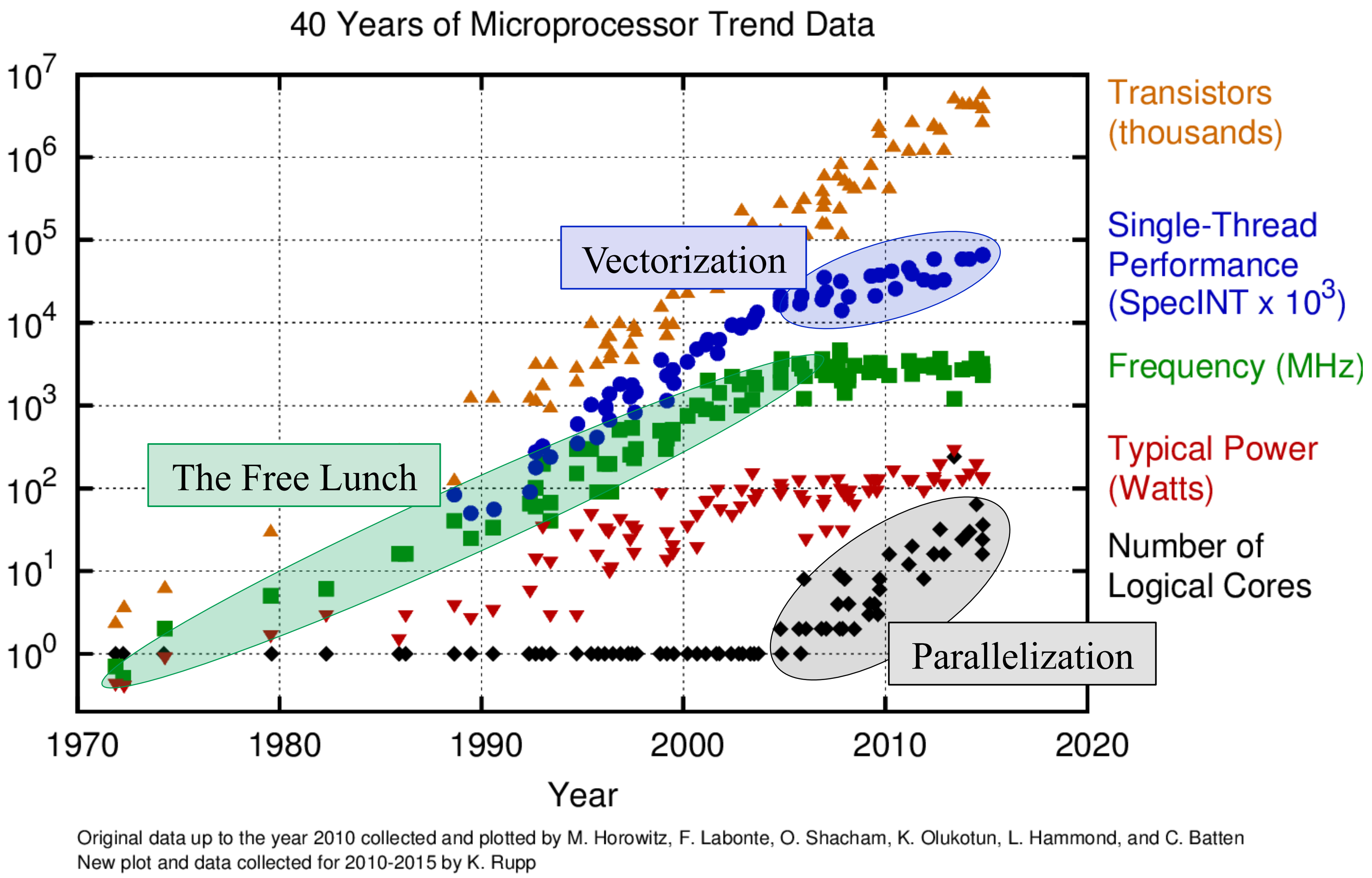
Moore’s Law

- Number of transistors doubles ever 18-24 months
- More transistors = better, right?

Microprocessor Transistor Counts 1971-2011 & Moore's Law



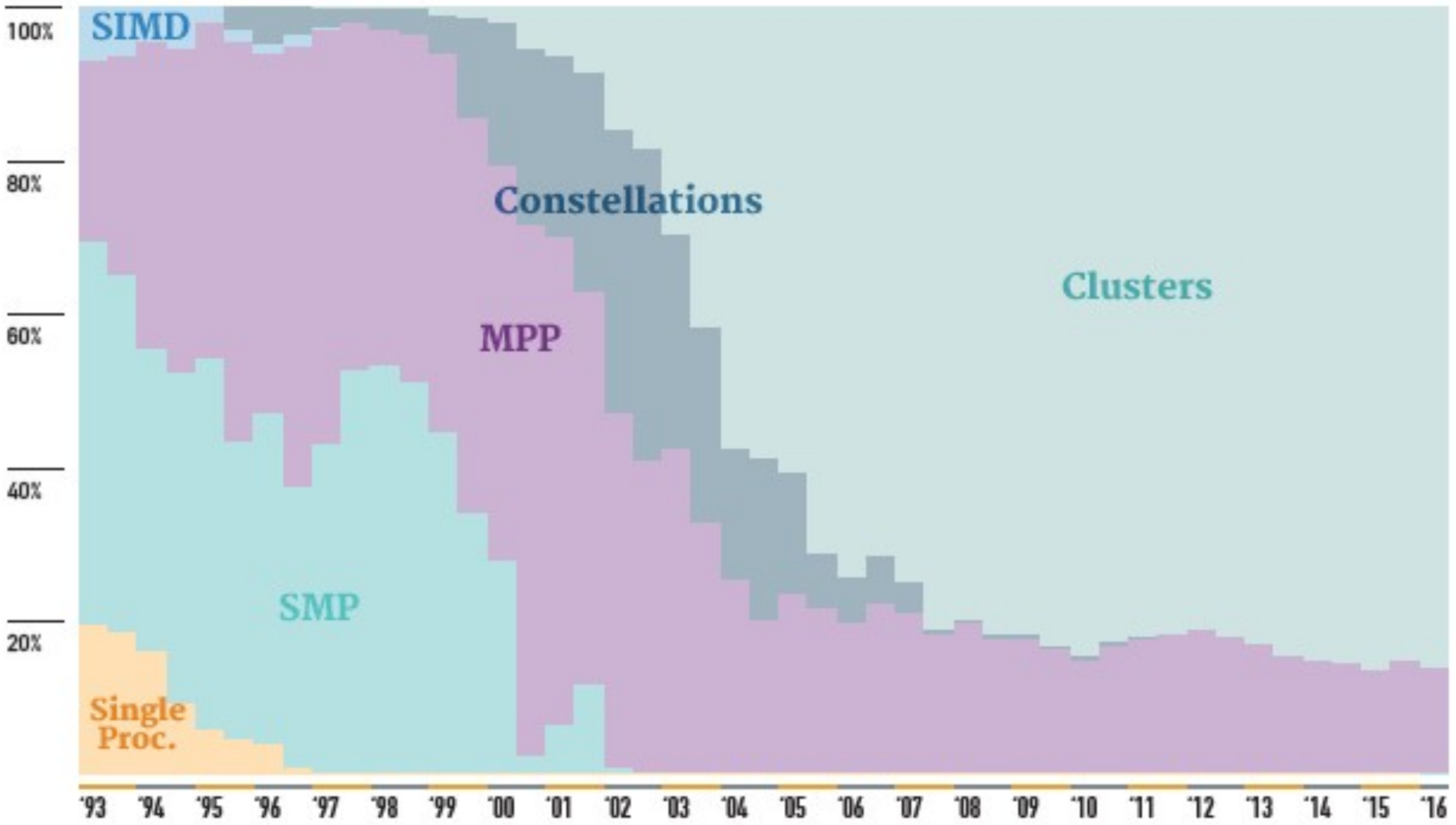
The Free Lunch is Over



https://www.karlrupp.net/2015/06/40-years-of-microprocessor-trend-data/

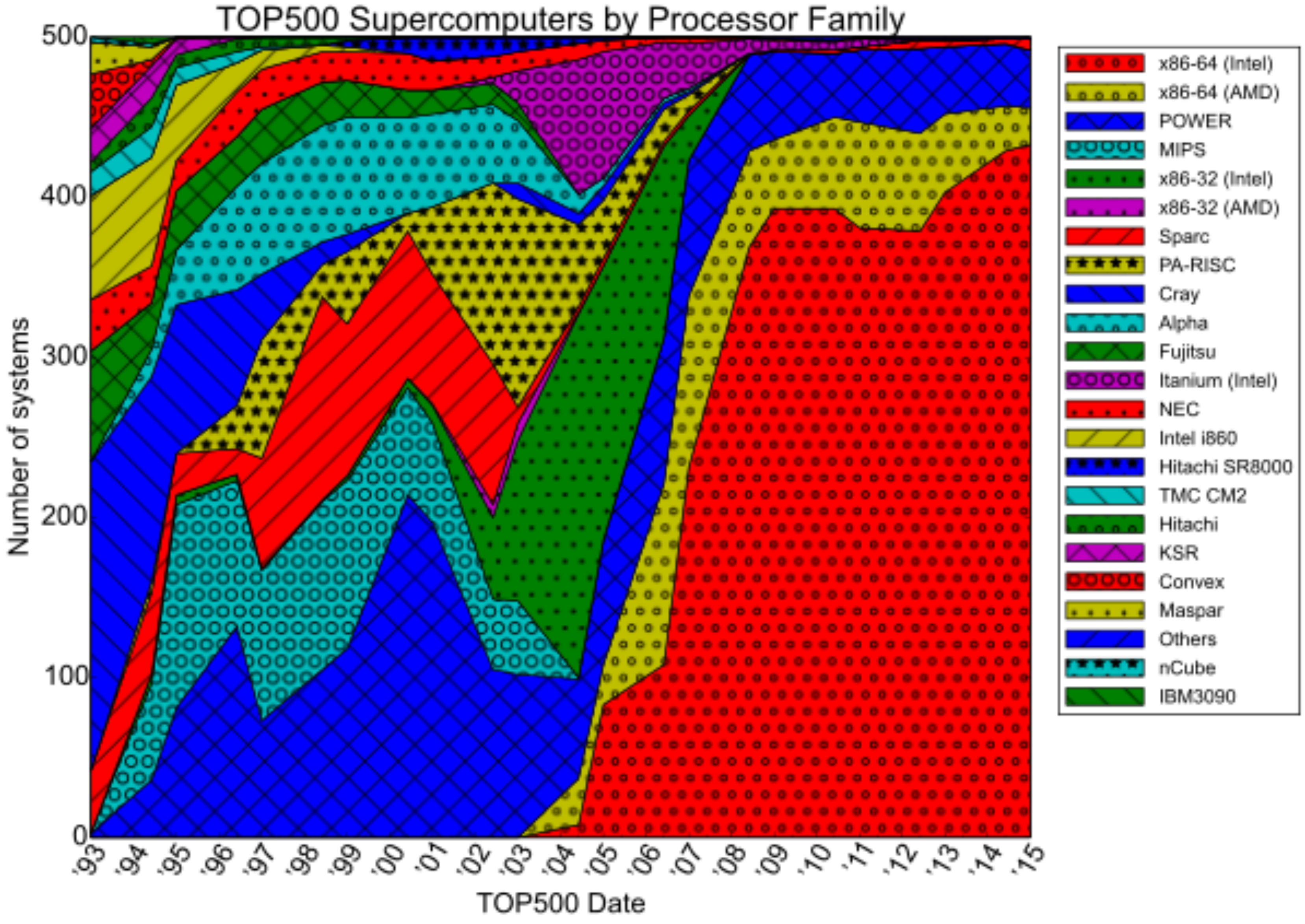
Top 500 Supercomputers

https://www.top500.org



https://www.nextplatform.com/2016/06/20/china-topples-united-states-top-supercomputer-user/

Intel Is (the Current) King



https://www.revolvy.com/page/TOP500

Outline

- Motivation
- HPC Cluster
- Compute
- Memory
- Disk
- Emerging Architectures

Zoom In On A Cluster

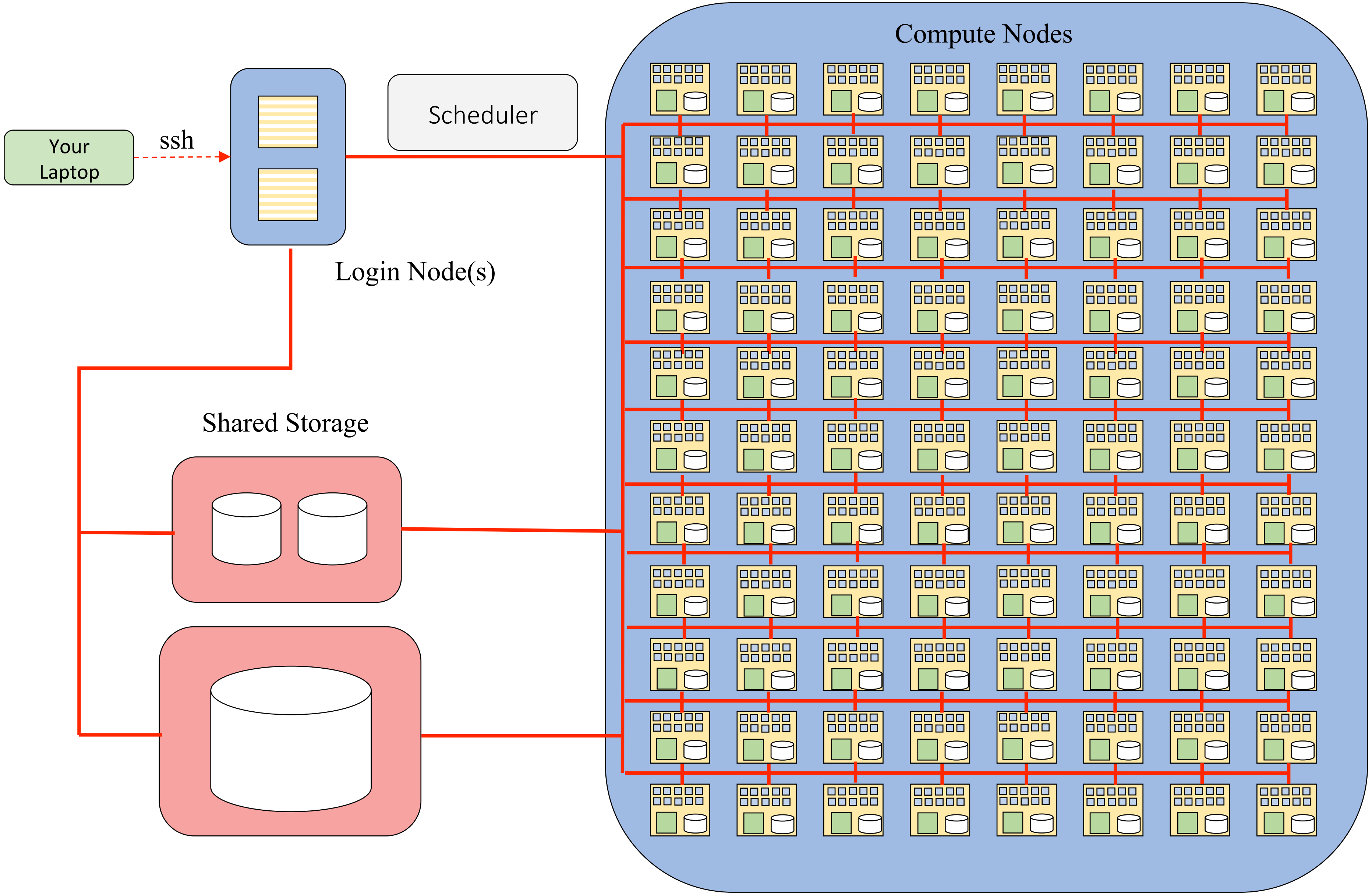
- A <u>cluster</u> is just *connected* collection of computers called <u>nodes</u>
- A single-unconnected node is usually called a <u>server</u>
- Entire clusters and stand alone servers are sometimes referred to as <u>machines</u>

Parallel Computers

- HPC [Linux] Cluster
- A collection of servers connected to form a single entity
    - Each is essentially a self contained computer
    - OS with CPU, RAM, hard drive, etc.
- Stored in racks in dedicated machine rooms
- Connected together via (low latency) interconnects
  - Ethernet < Infiniband, OPA (Intel's Omi-Path Architecture)
- Connected to storage
- Vast majority of HPC clusters
- SMP System
  - Symmetric Multi-Processing
  - CPUs all share memory – essentially one computer
  - Expensive and serve unique purpose
    - Huge memory and huge OpenMP jobs

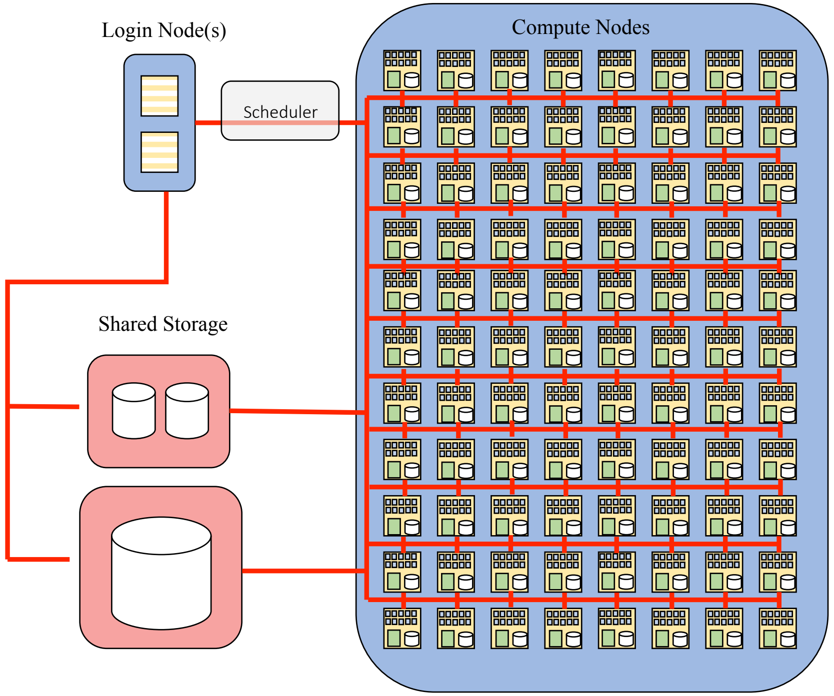
Princeton's Della Cluster

Basic Cluster Layout

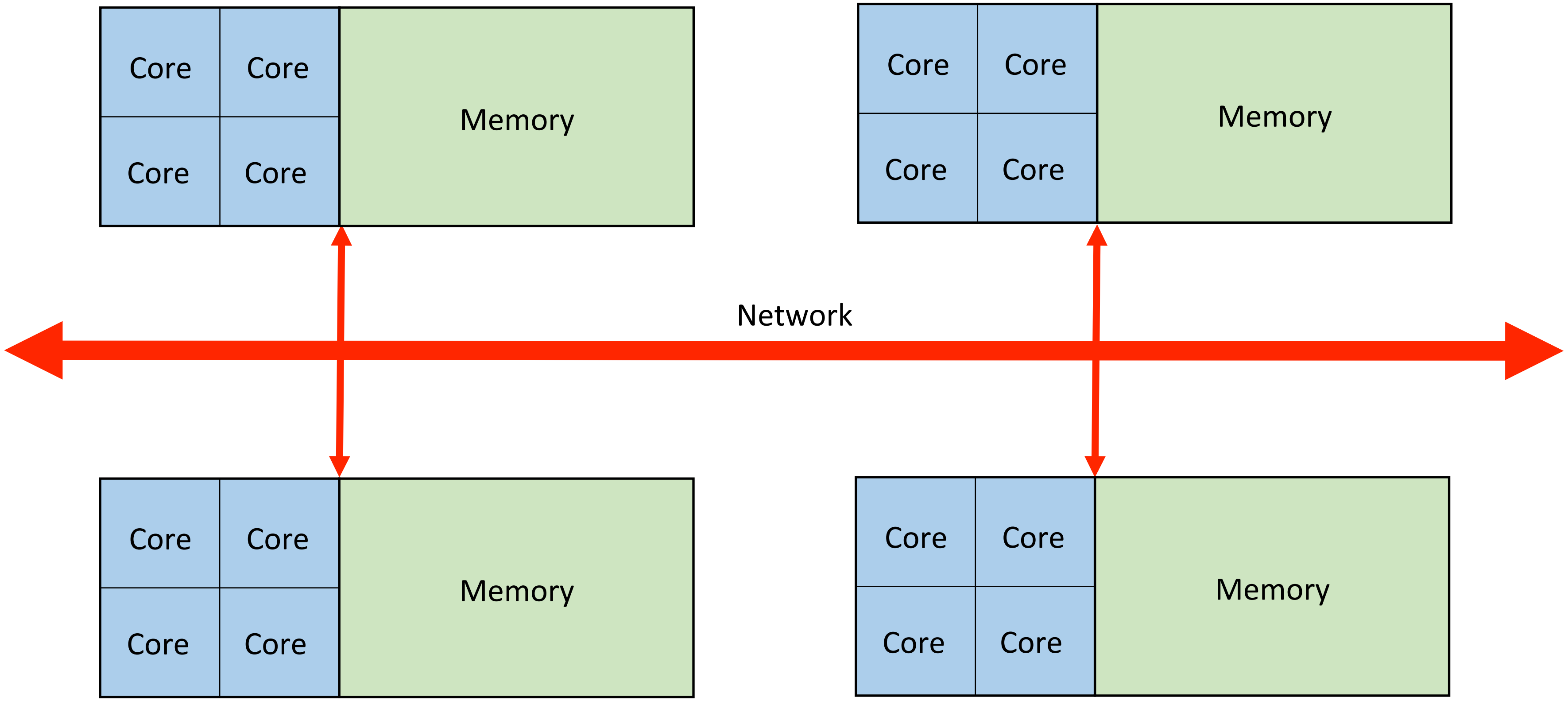


Cluster Decomposed

- Login Node(s)
  - Edit, debug, compile, and interact with scheduler
  - Not for long running jobs!
- Scheduler
  - You tell the scheduler what resources you need
    - # of cpus, #of nodes, GB of memory, # of hours, etc
    - Then what to do: “run this program”
  - Scheduler then assigns hardware exclusive for your job
  - SLURM on our machines
- Storage
  - /home directories NFS mounted everywhere
  - Each compute node has local storage
  - Some clusters have more storage options
    - Parallel, backed up long term storage
    - See researchcomputing.princeton.edu



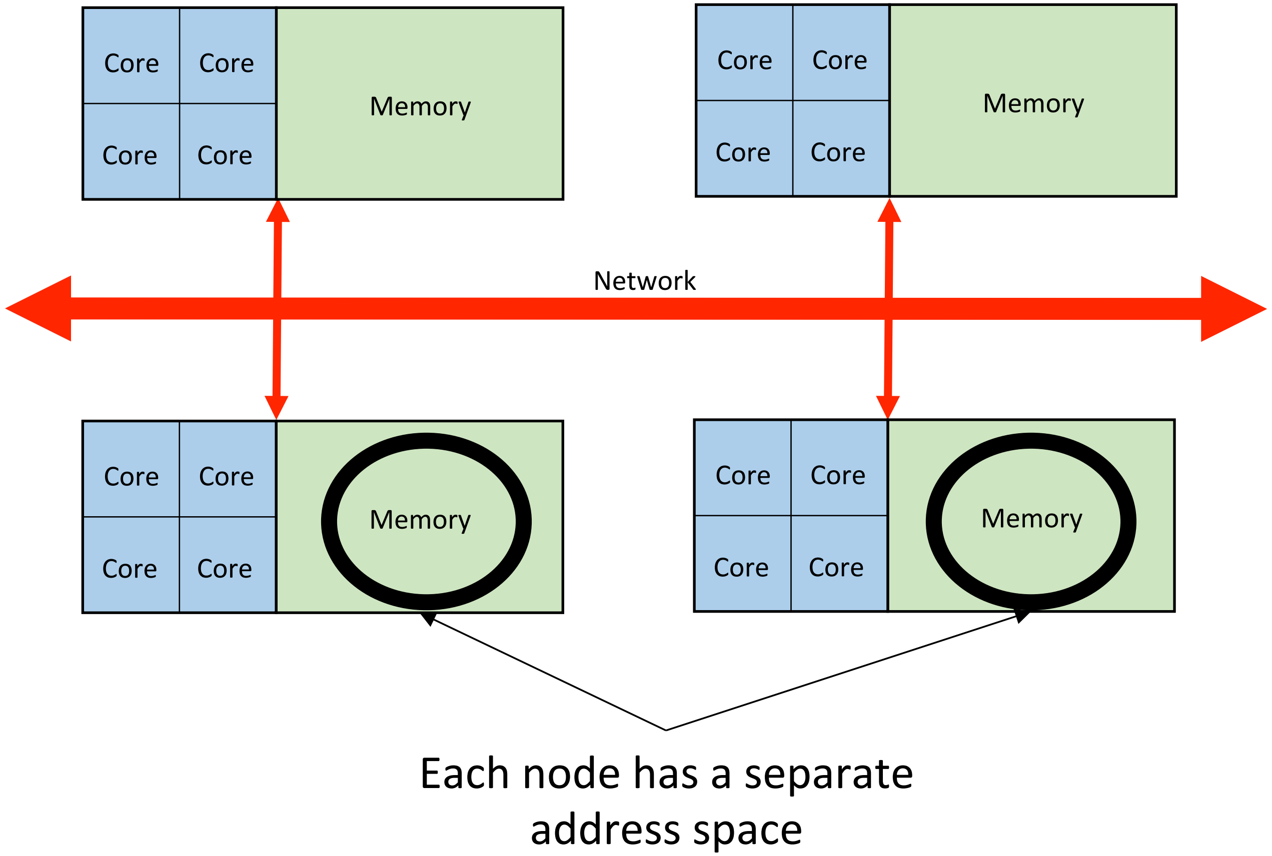
Modern HPC Cluster



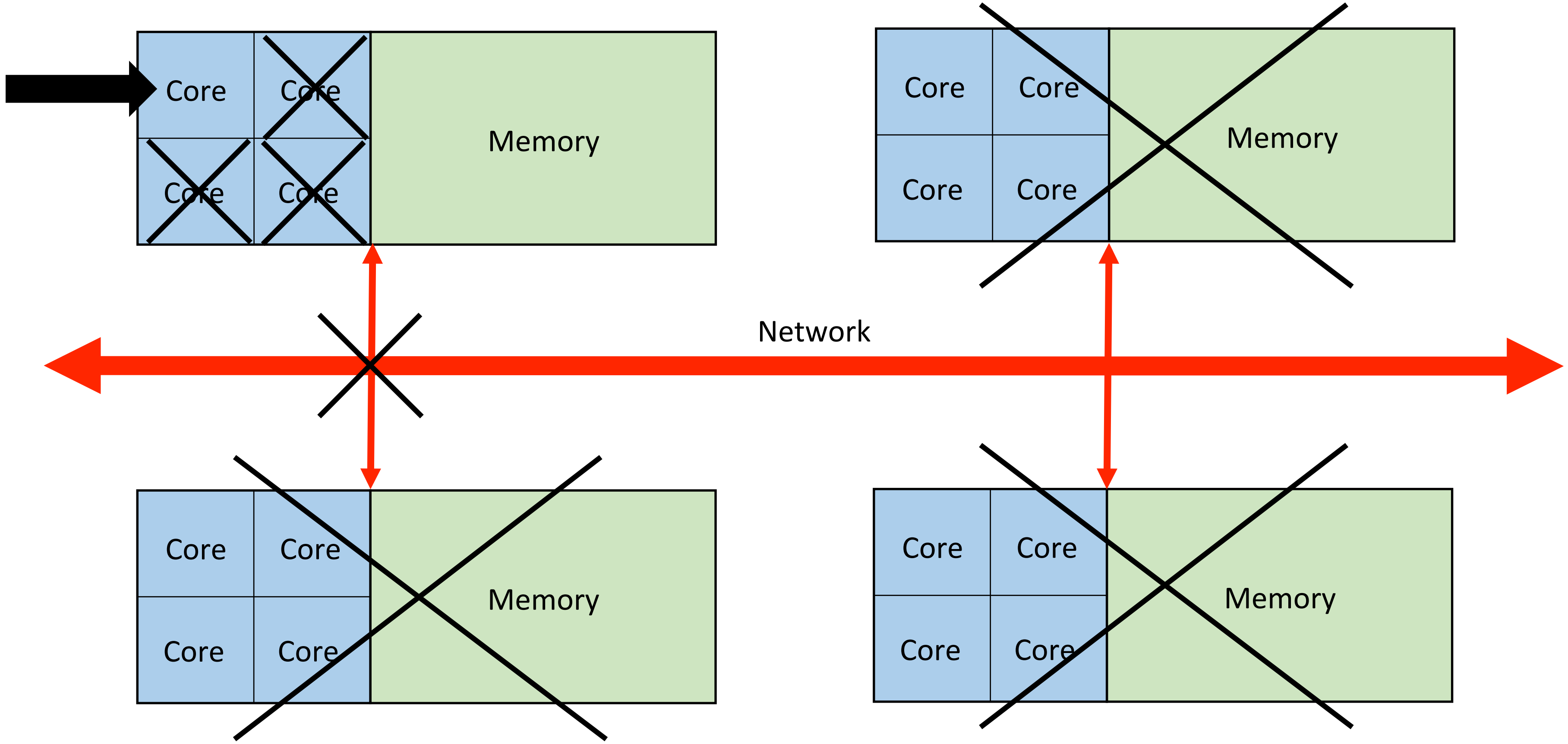
Tiger has 40 cores per node and 392 nodes
For a total of more than 15,600 cores!

Nodes Are Discrete Systems

- Memory Address space
  - Range of unique addresses
- Separate for each node
- Can't just access all memory in cluster

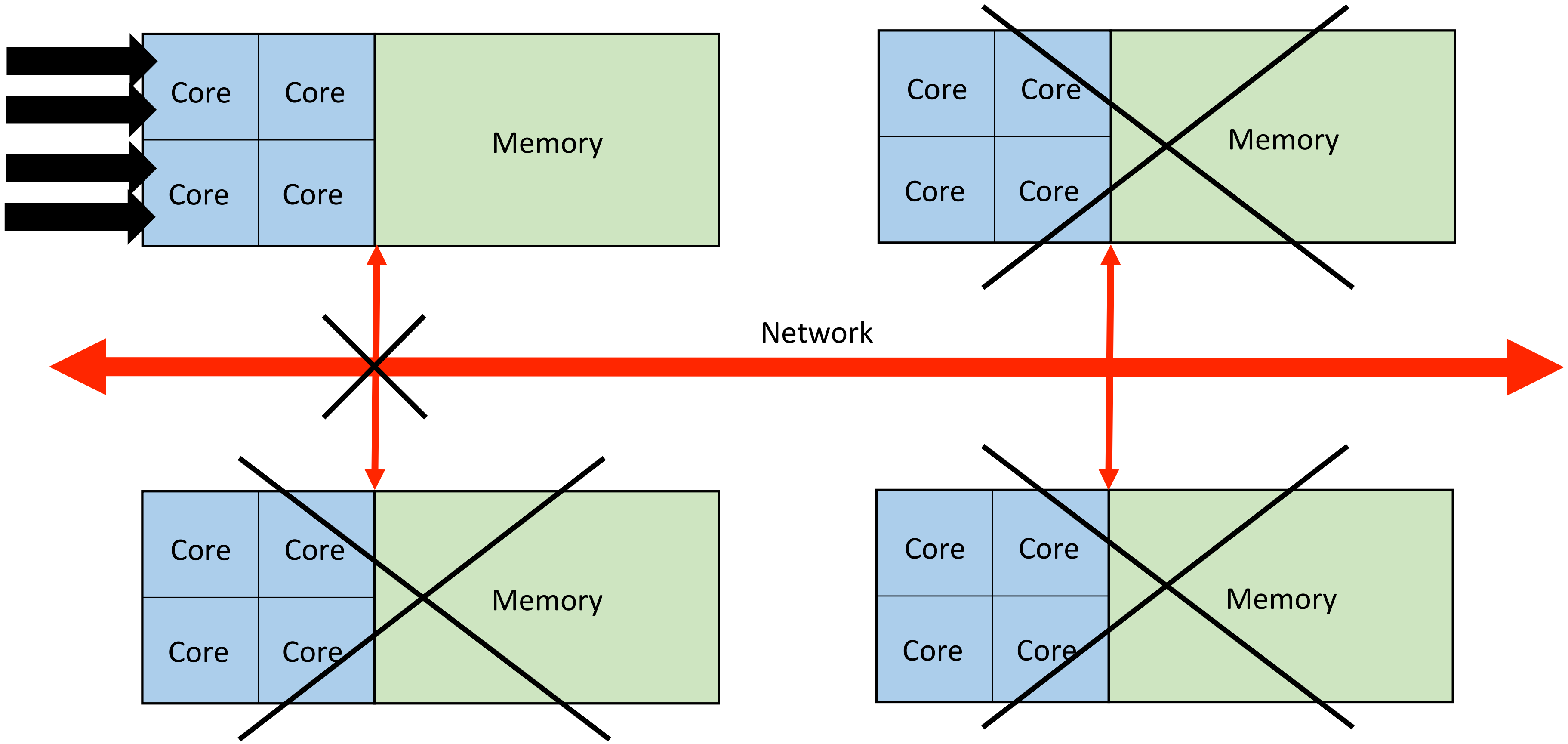


Serial Code

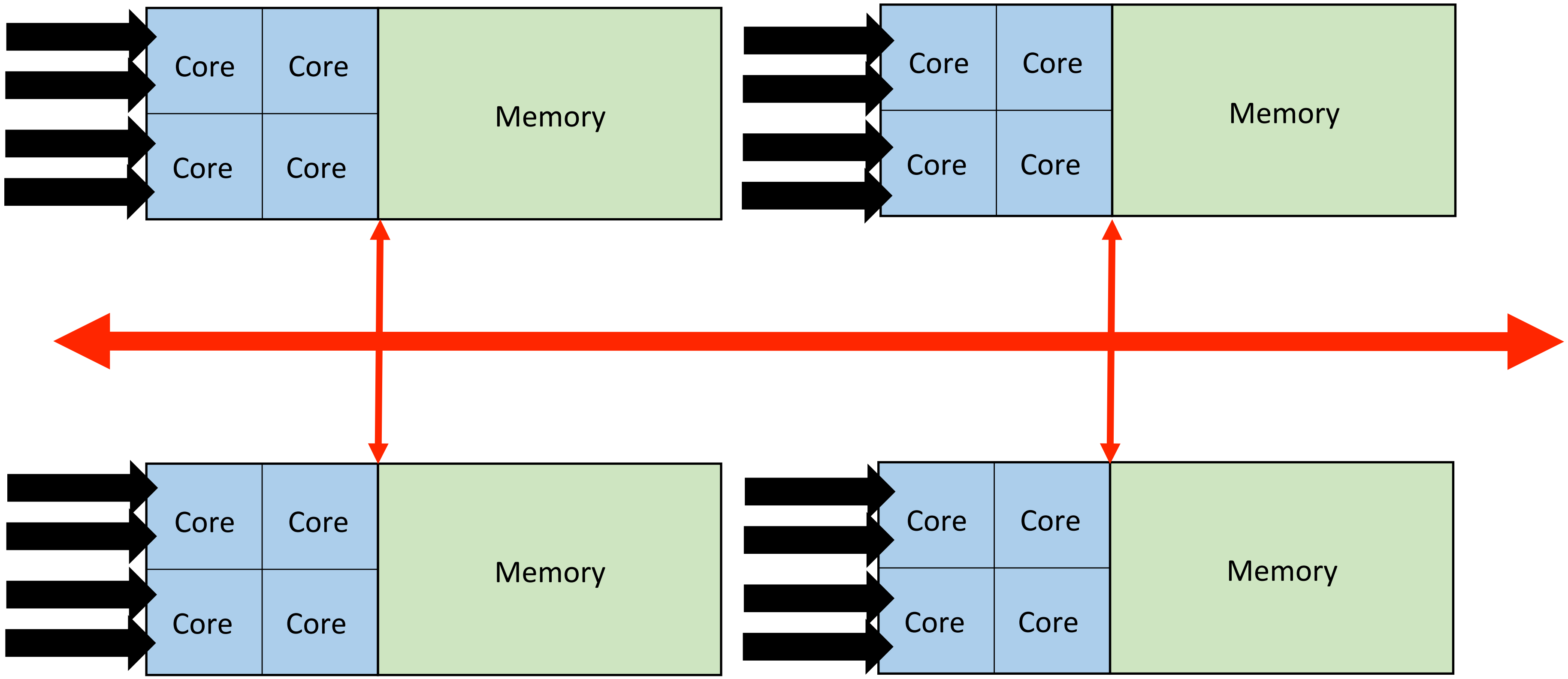


Core
Core
Core
Core
Memory
Core
Core
Core
Core
Memory
Network
Core
Core
Core
Core
Memory
Core
Core
Core
Core
Memory

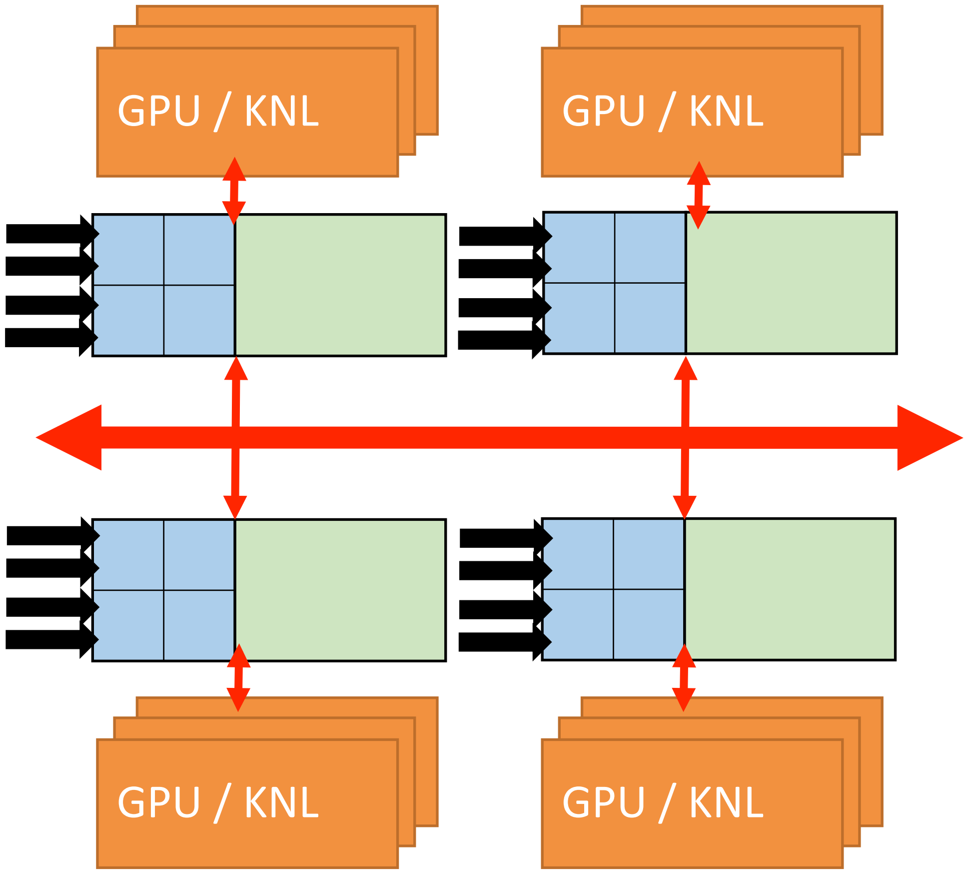
Multithreading (OpenMP, Pthreads, etc.)



Message Passing Interface (MPI)



Clusters With Accelerators



- Accelerators: GPUs or Xeon Phi (KNC, KNL)
- Programmable with MPI + x
  - Where x = OpenMP, OpenACC, CUDA, …
  - Or x = OpenMP + CUDA, …
- Increase computational power
  - Increase FLOPS / Watt
- Top500.org
  - Shift toward systems with accelerators a few years ago

Outline

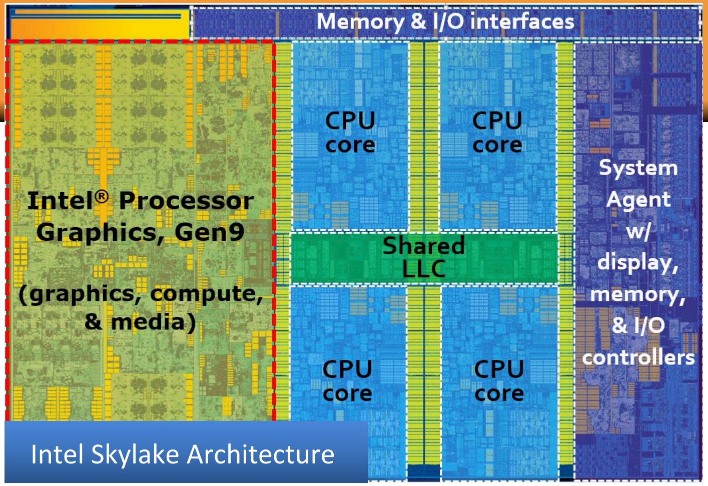
- Motivation
- HPC Cluster
- Compute
- Memory
- Disk
- Emerging Architectures

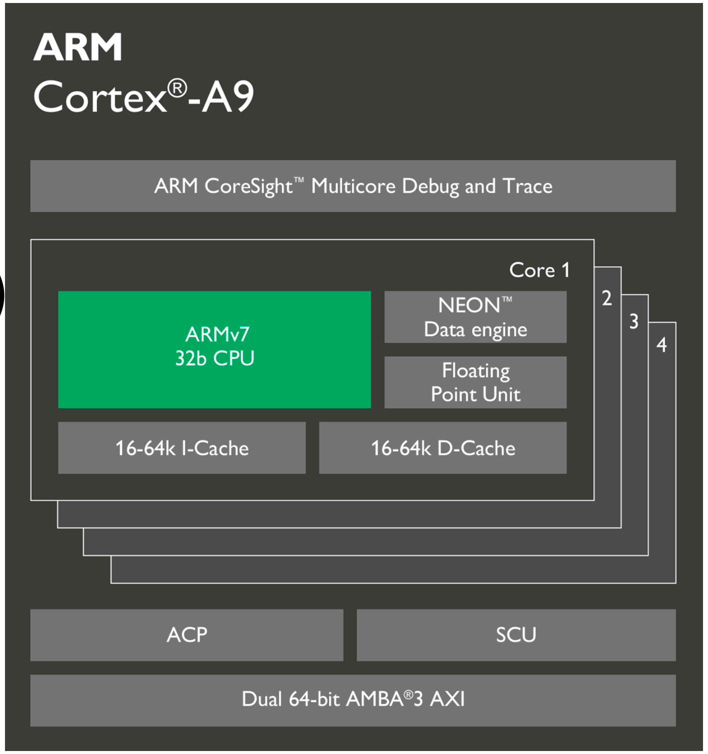
Multicore Processors

- (Most) modern processors are multicores
  - Intel Xeon
  - IBM Power
  - AMD Opteron
  - ARM processors
  - Mobile processors

- Most have vector units
- Most have multiple cache levels
- Require special care to program (multi-threading, vectorization, …)

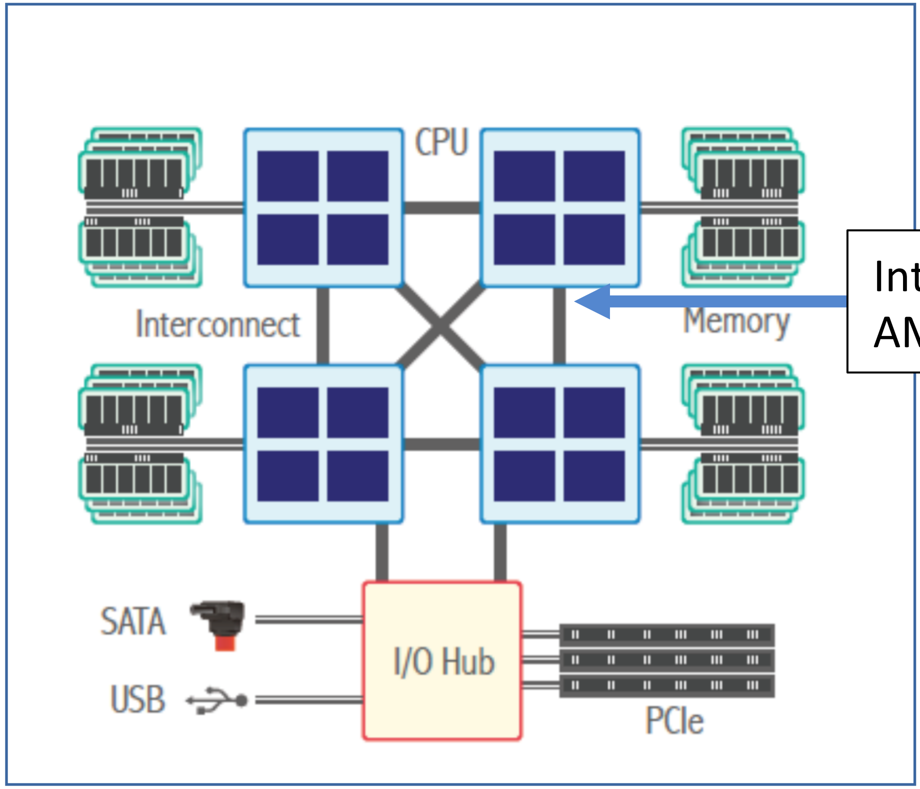
- Instruction set
  - x86_64 – Intel and AMD
  - Power, ARM, NVIDIA, etc have different instruction sets





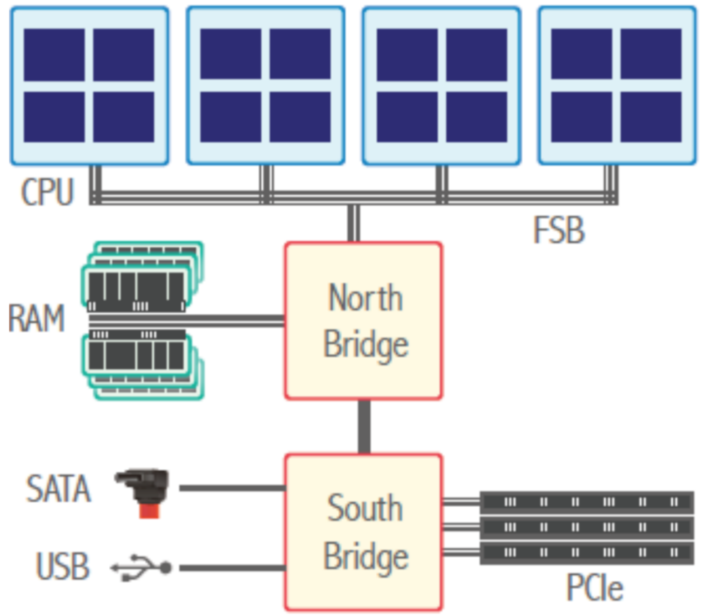
Motherboard Layout

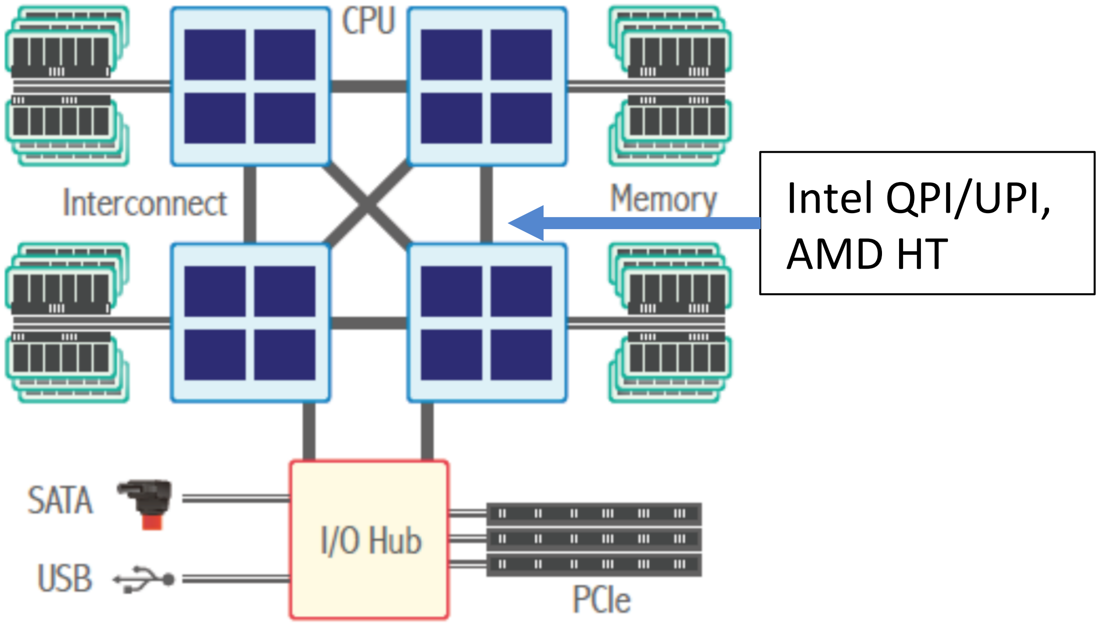
- Typical nodes have 1, 2, or 4 CPUs
  - 1: laptop/desktops/servers
  - 2: servers/clusters
  - 4: high end special use
- Each CPU is attached to a socket
- Connected together by interconnect
- Memory associated with CPU
  - Programmer sees single node memory address space
- Each CPU has multiple cores



Intel QPI/UPI, AMD HT

Motherboard Layouts

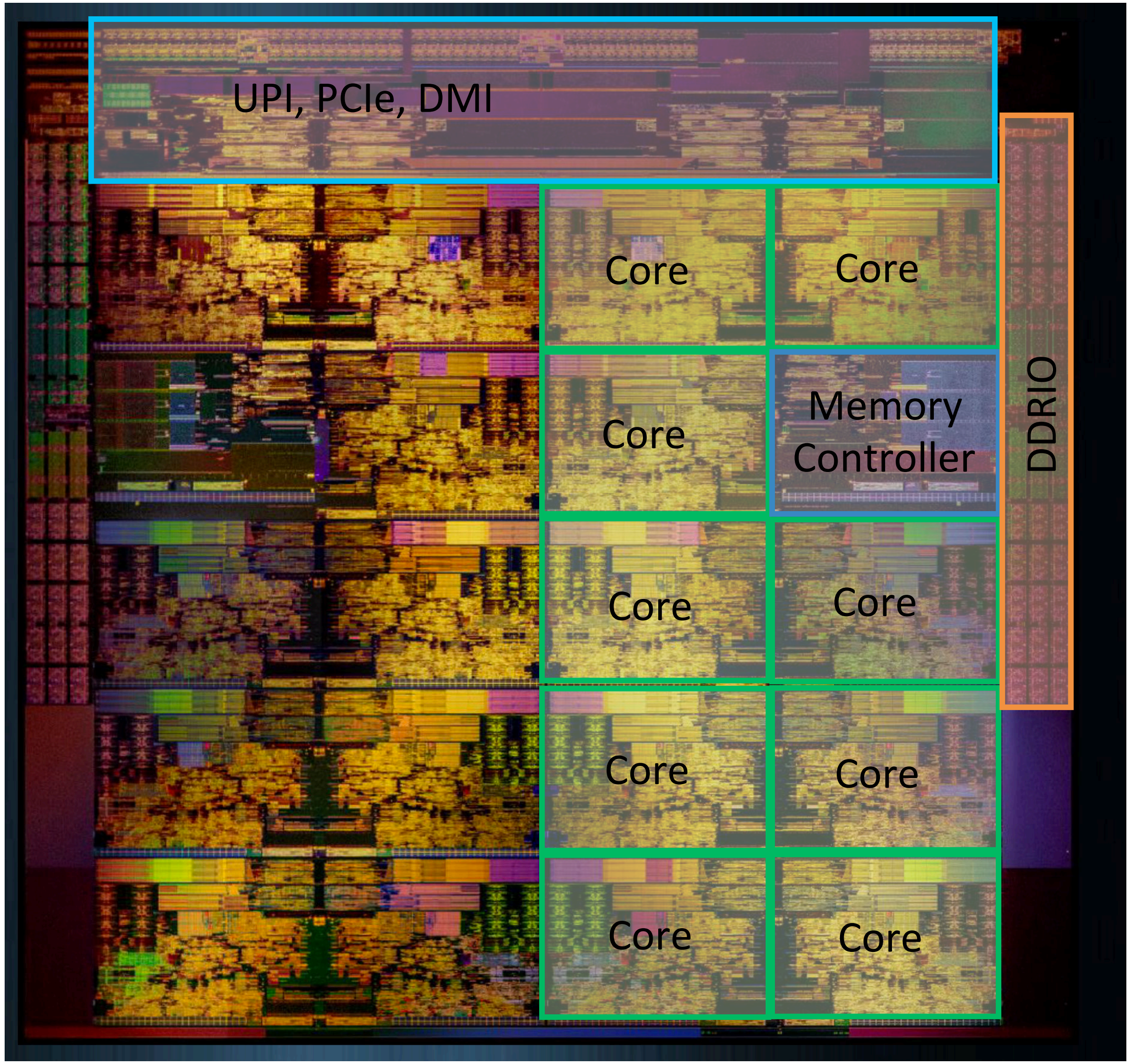


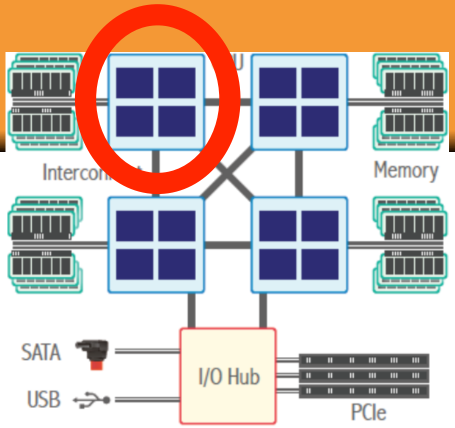


Symmetric MultiProcessing (SMP)
front side bus (FSB) bottlenecks in SMP systems, device-to-memory bandwidth via north bridge, north bridge to RAM bandwidth limitations.

Non-Uniform Memory Access (**NUMA**)
physical memory partitioned per CPU, fast interconnect to link CPUs to each other and to I/O. Remove bottleneck but memory is no longer uniform – 30-50% overhead to accessing remote memory, up to 50x in obscure multi-hop systems.

Example: Intel Skylake CPU





Core Definition (Jarp):

- "A complete ensemble of execution logic, and cache storage as well as register files plus instruction counter (IC) for executing a software process or thread."

Memory Controller

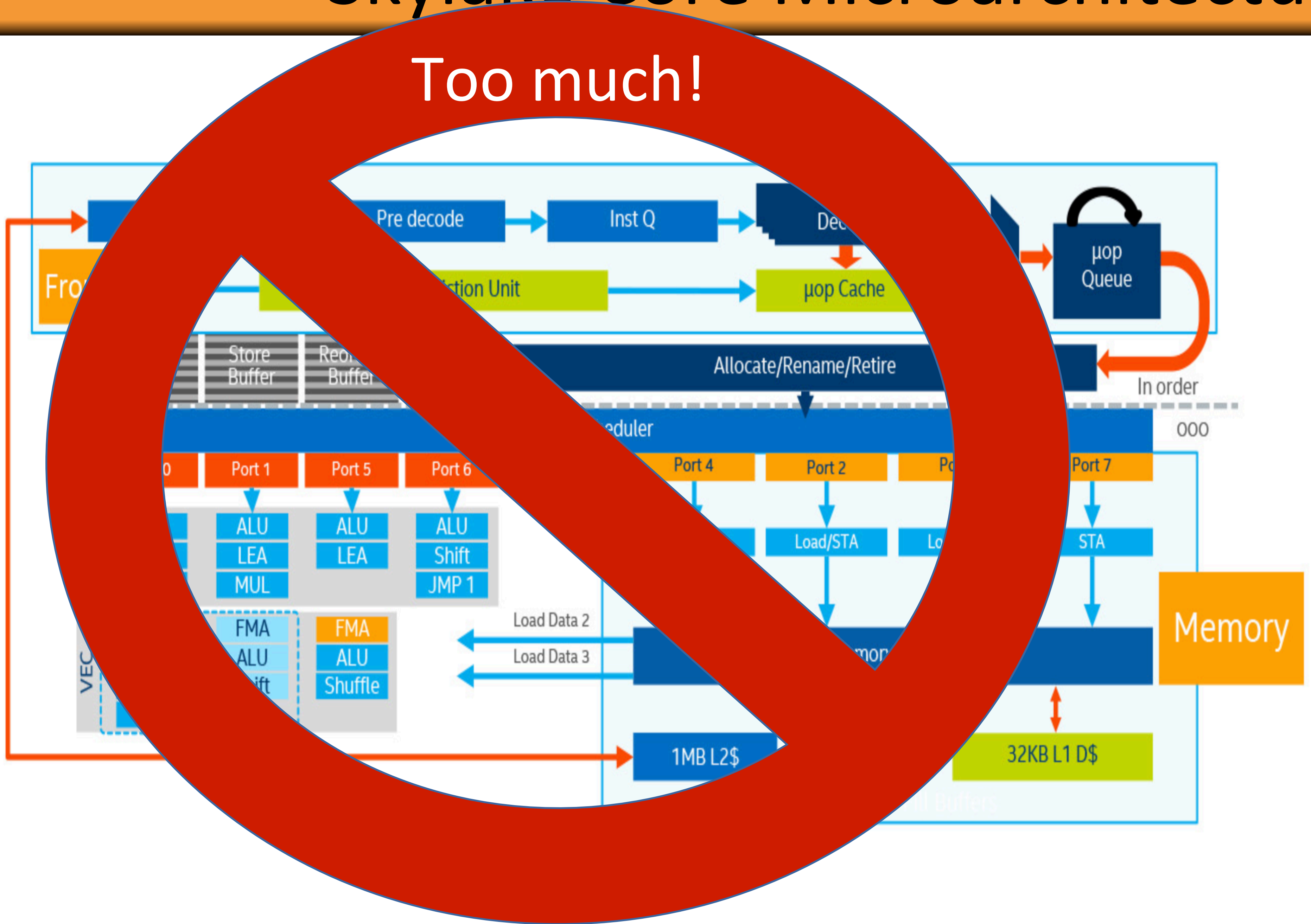
UPI – Intel UltraPath Interconnect

PCIe – Peripheral Component Interconnect Express

DMI – Direct Media Interface

DDRIO – Double Data Rate IO

Skylake Core Microarchitecture



Key Components:

Control logic
Register file
Functional Units

- ALU (arithmetic and logic unit)
- FPU (floating point unit)

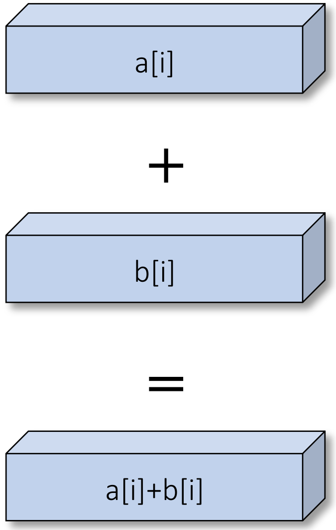
- Data Transfer
- Load / Store

Vectorization

At the inner-most loop level

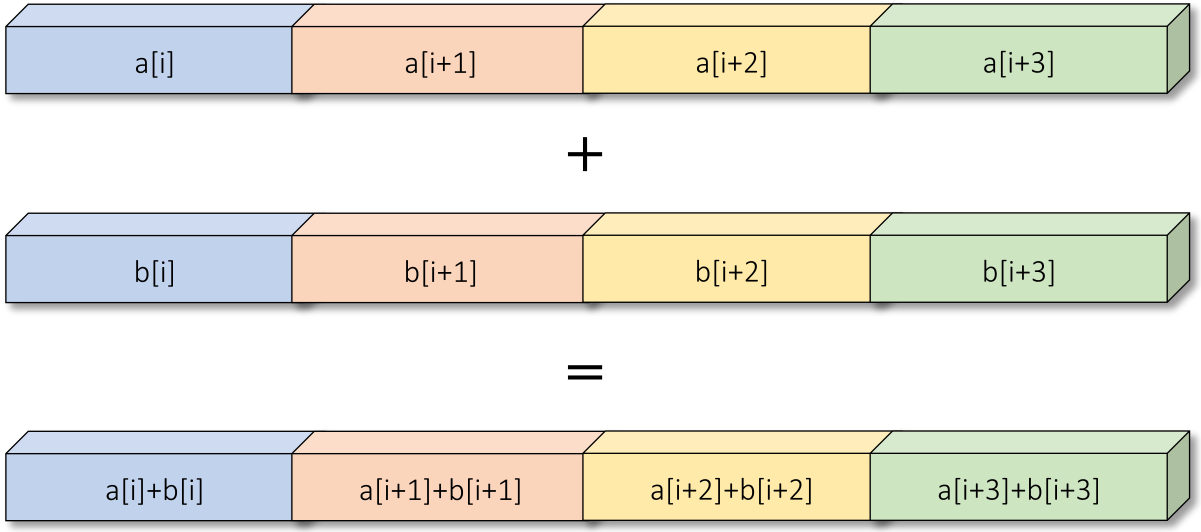
```
for (int i=0; i<N; i++) {
   c[i]=a[i]+b[i];   }
```

Scalar (SISD)



One operation
One result

Vector (SIMD)



One operation
Multiple results

Vectorization Terminology

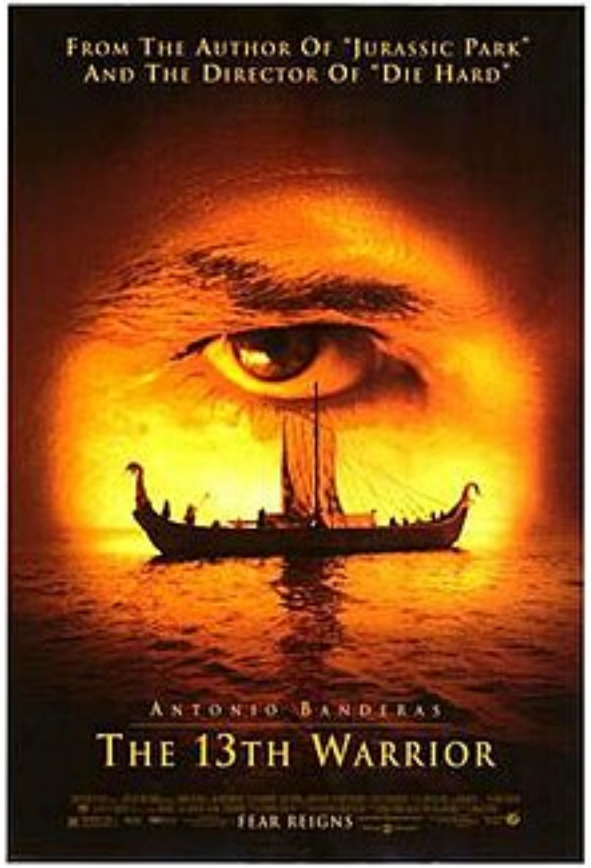
- SIMD: Single Instruction Multiple Data
- New generations of CPUs have new capabilities
  - Machine (assembly) commands that take advantage of new hardware
  - Called “instruction set”
- SSE: Streaming SIMD Extensions
- AVX: Advanced Vector Extension

HPC Performance Metrics

- FLOP = Floating-point Operation
- FLOPs = Floating-point Operations
- FLOPS = Floating-point Operations Per Second
  - Often FLOP/sec to avoid ambiguity

- FLOPS/$
- FLOPS/Watt
- Bandwidth: GB/s

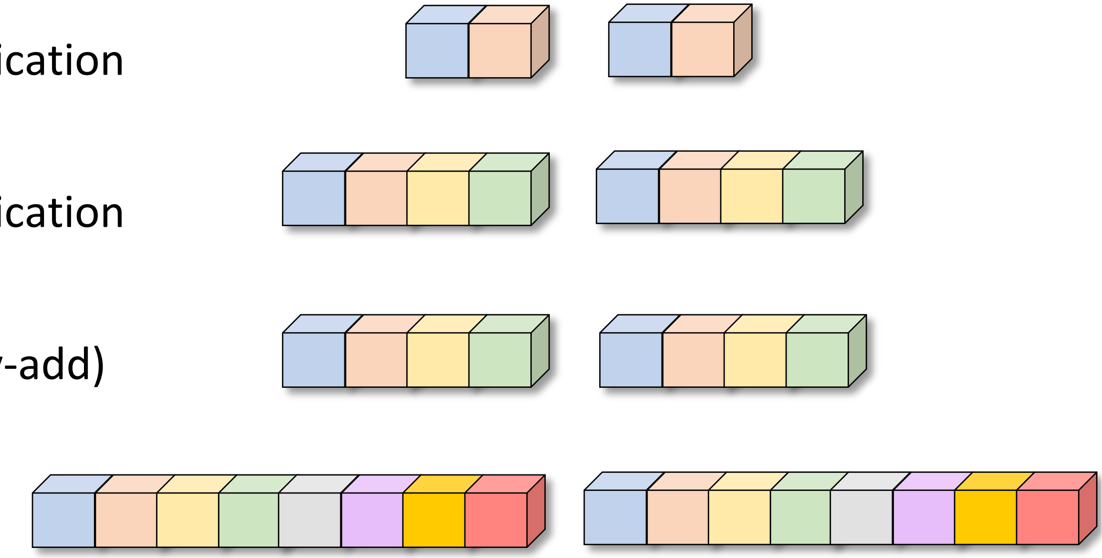
- Usually Double Precision (64-bit)
- Calculation
  - FLOPS = (Clock Speed) *(Cores)*(FLOPs/cycle)



N.B. - not to be confused with a Hollywood flop!

Intel Xeon (Server) Architecture Codenames

- Number of FLOP/s depends on the chip architecture
- Double Precision (64 bit double)
  - Nehalem/Westmere (SSE):
    - 4 DP FLOPs/cycle: 128-bit addition + 128-bit multiplication
  - Ivybridge/Sandybridge (AVX)
    - 8 DP FLOPs/cycle: 256-bit addition + 256-bit multiplication
  - Haswell/Broadwell (AVX2)
    - 16 DP FLOPs/cycle: two, 256-bit FMA (fused multiply-add)
  - KNL/Skylake (AVX-512)
    - 32 DP FLOPs/cycle: two, 512-bit FMA



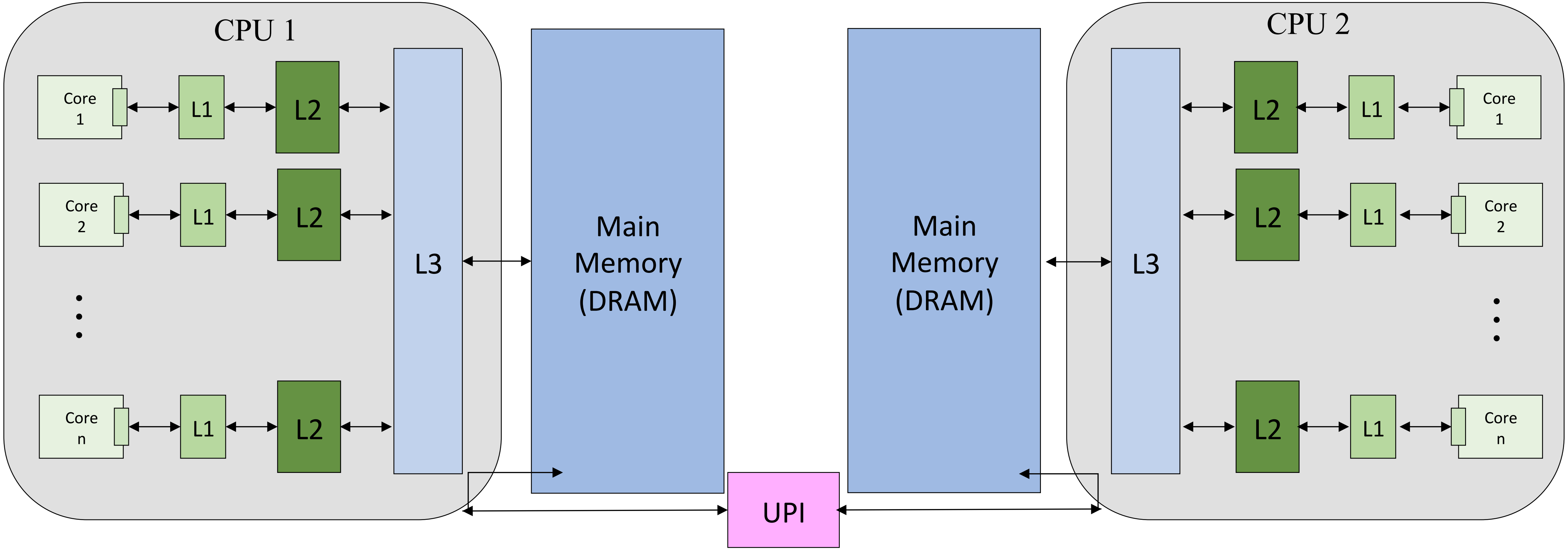
- FMA = (a × b + c)
- Twice as many if single precision (32-bit float)

Outline

- Motivation
- HPC Cluster
- Compute
- Memory
  - Layout
  - Cache
  - Performance
- Disk
- Emerging Architectures

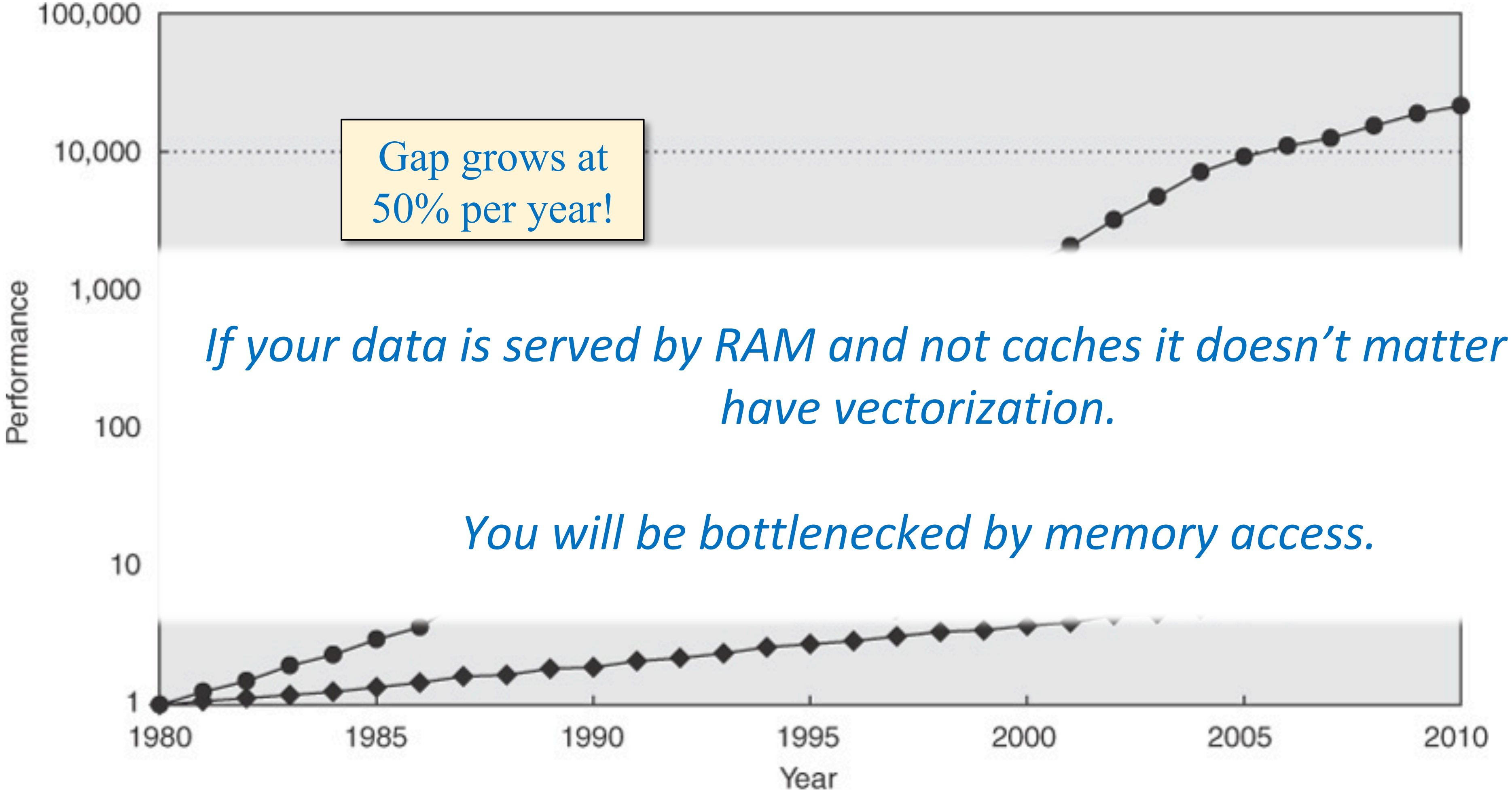
Memory Hierarchy

Dual Socket Intel Xeon CPU



	Registers	L1 Cache	L2 Cache	L3 Cache	DRAM	Disk
Speed	1 cycle	~4 cycles	~10 cycles	~30 cycles	~200 cycles	10ms
Size	< KB per core	~32 KB per core	~256 KB per core	~35 MB per socket	~100 GB per socket	TB

Does It Matter?



If your data is served by RAM and not caches it doesn't matter if you have vectorization.

You will be bottlenecked by memory access.

http://web.sfc.keio.ac.jp/~rdv/keio/sfc/teaching/architecture/architecture-2008/hennessy-patterson/Ch5-fig02.jpg

Outline

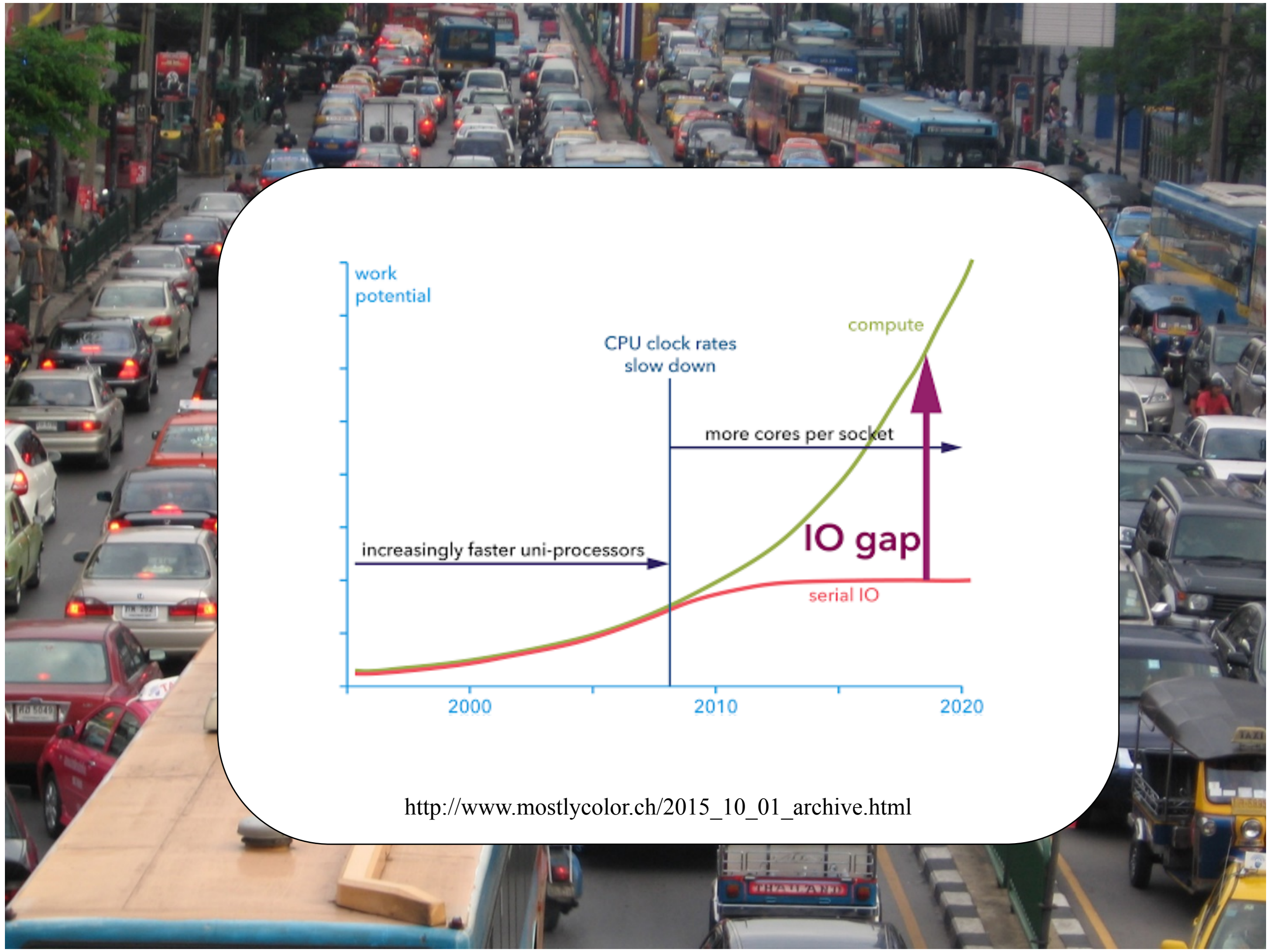
- Motivation
- HPC Cluster
- Compute
- Memory
- Disk
  - Types
  - Filesystems
- Emerging Architectures

File I/O

"A supercomputer is a device for turning compute-bound problems into I/O-bound problems."

-- *Ken Batcher, Emeritus Professor of Computer Science at Kent State University*

File I/O?



https://en.wikipedia.org/wiki/Transport_in_Bangkok

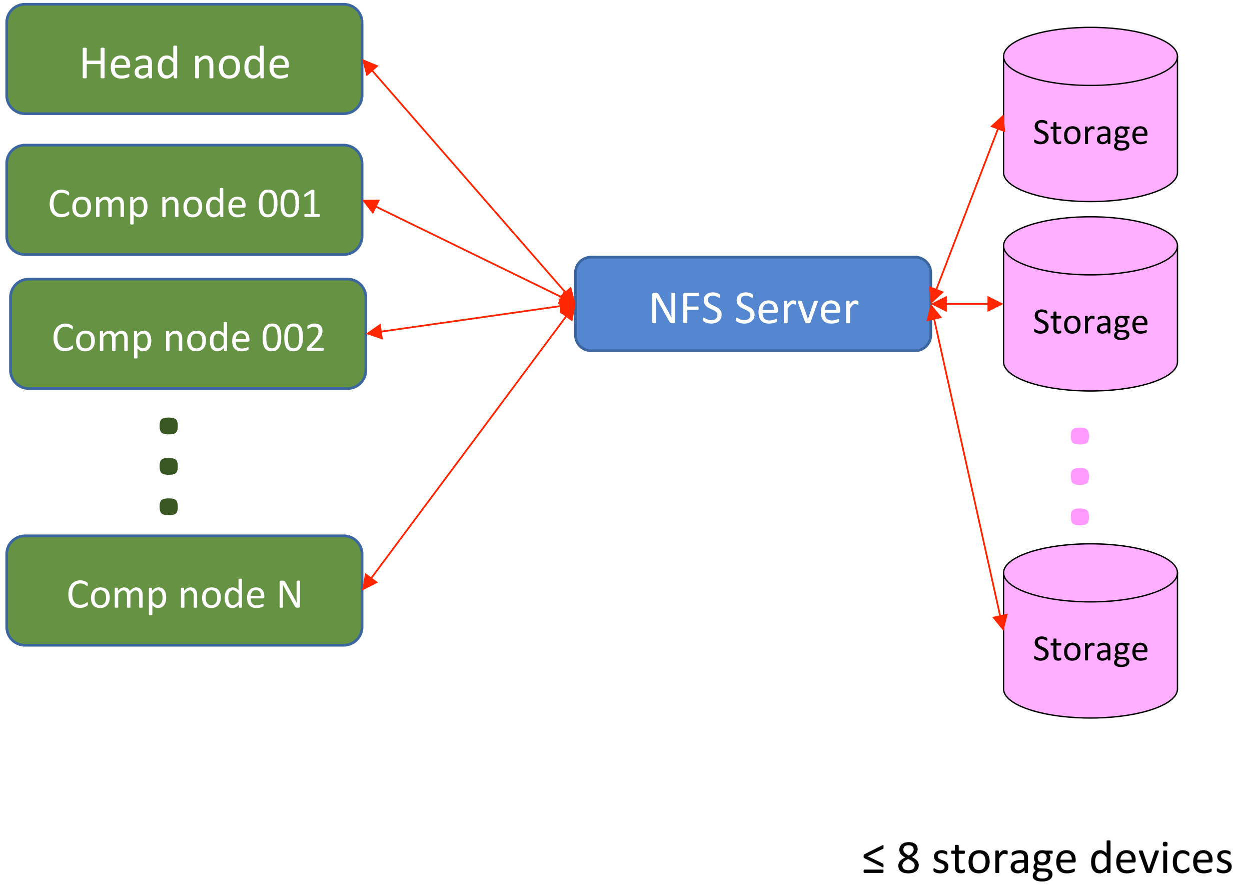
Types of Disk

- Hard Disk Drive (HDD)
  - Traditional Spinning Disk
- Solid State Drives (SSD)
  - ~5x faster than HDD
- Non-Volatile Memory Express (NVMe)
  - ~5x faster than SSD

Photo Credit: Maximum PC

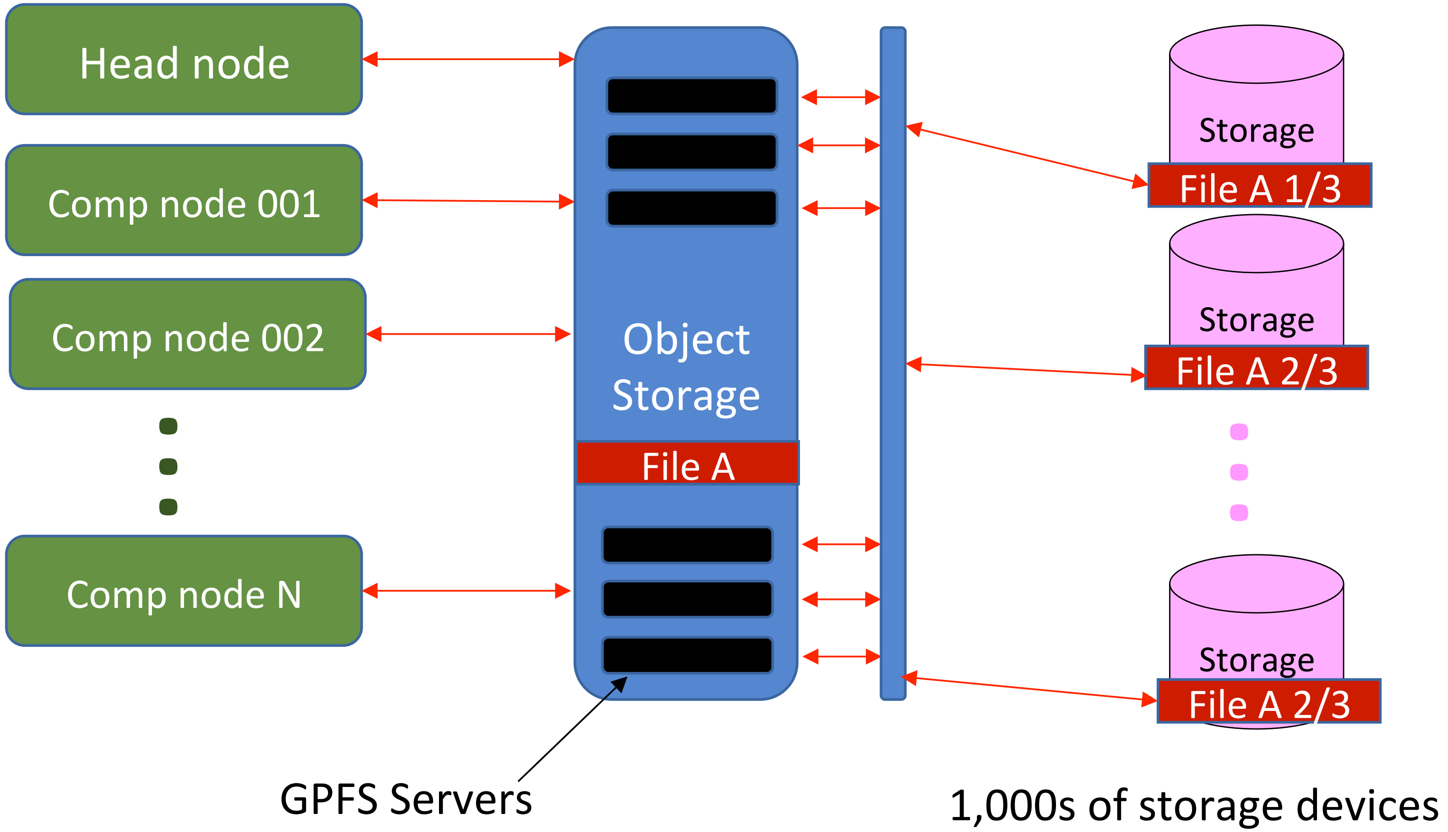
Types of Filesystems - NFS

- NFS – Network File System
  - Simple, cheap, and common
  - Single filesystem across network
  - Not well suited for large throughput
  - (/home on Princeton clusters)



Parallel Filesystems

- GPFS (General Parallel Filesystem – IBM)
  - Designed for parallel read/writes
  - Large files spread over multiple storage devices
  - Allows concurrent access
  - Significantly increase throughput
  - (/tigress and /scratch/gpfs)
- Lustre
  - Similar idea
  - Different implementation
- Parallel I/O library in software
  - Necessary for performance realization

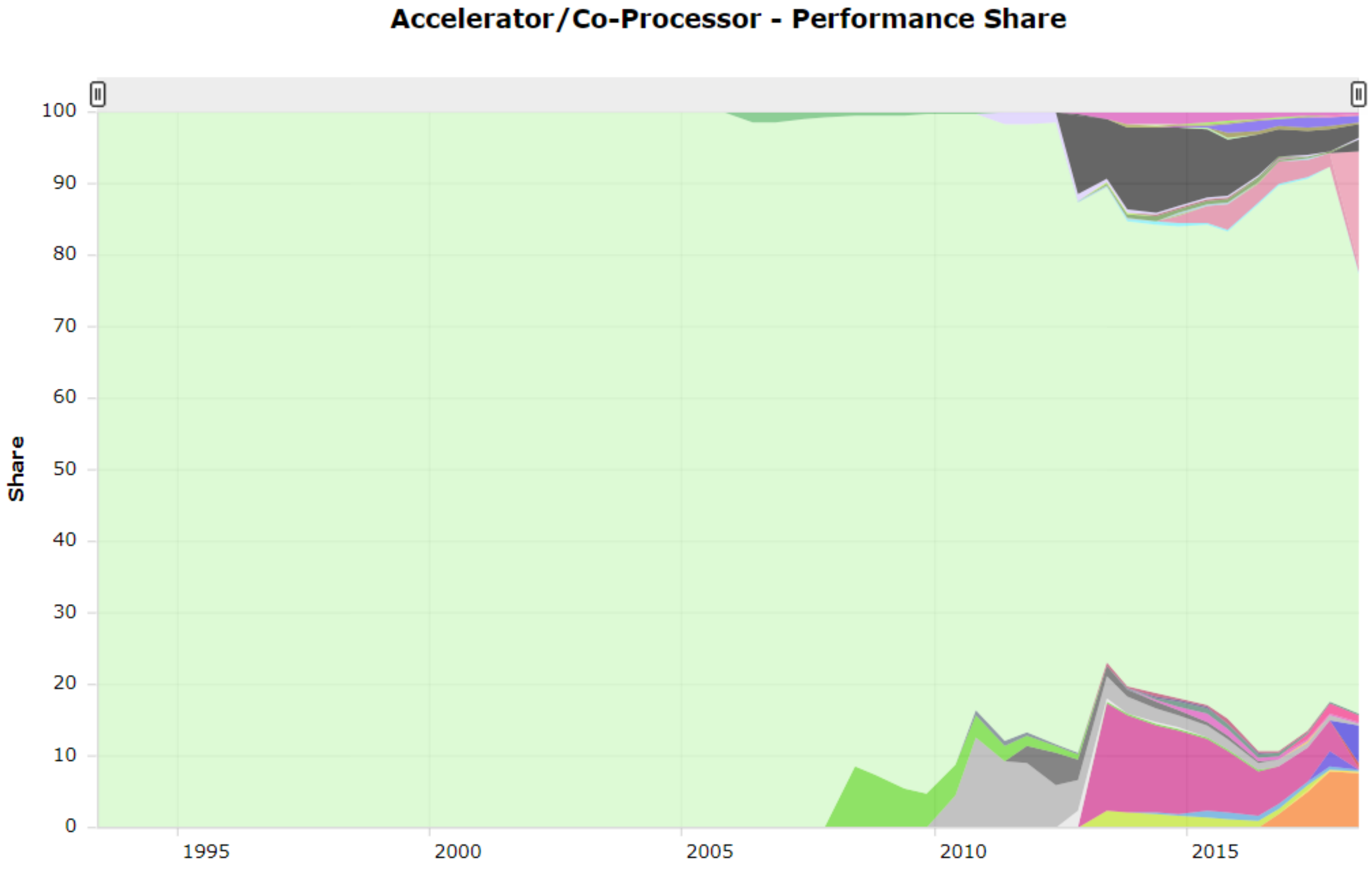


Outline

- Motivation
- HPC Cluster
- Compute
- Memory
- Disk
- Emerging Architectures
  - Xeon Phi
  - GPU
  - Cloud

Emerging Architectures

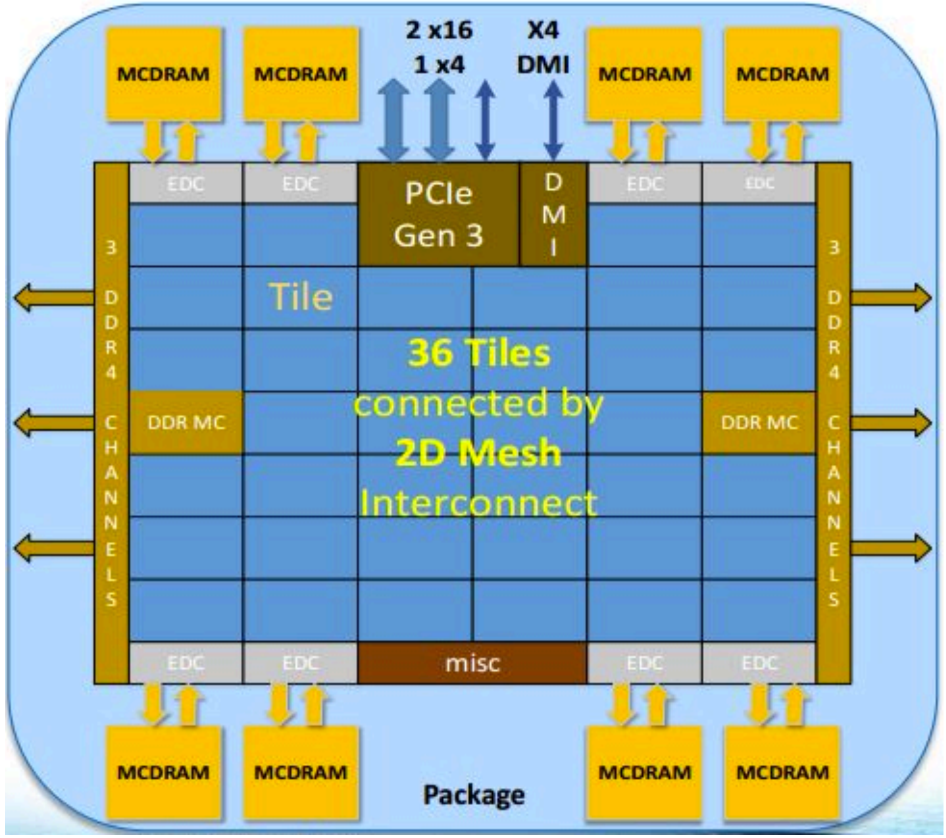
- The computing landscape changes rapidly
- 110 TOP500 systems have accelerators (June 2018)



https://www.top500.org/statistics

Intel Xeon Phi a.k.a. Intel MIC

- MIC: Many Integrated Cores
- x86-compatible multiprocessor architecture
- Mostly used as a co-processor
  - Data travels over PCIe
- Programmable using
  - C, C++, Fortran
  - MPI, OpenMP, OpenCL, …
- ~1 year ago
  - Intel announced end of product line



NVIDIA Pascal

- Graphic Processing Unit (GPU)
- Exclusively a co-processor
  - Data travels over PCIe
- Not compatible with x86 library
- Programmable using:
  - Cuda, OpenCL
  - OpenACC, OpenMP

Accelerator Comparison

Lefebvre

	Xeon Gold 5122	Xeon Platinum 8180M	**Xeon Phi 7290F**	**NVIDIA V100**
Cores	4	28	72	84 SMs
Logical Cores	8	56	288	5120 cores
Clock rate	3.6 – 3.7 GHz	2.5 – 3.8 GHz	1.5-1.7 GHz	1530 MHz
Theoretical GFLOPS (double)	460.8	2240	3456	7450
SIMD width	512 bit	512 bit	512 bit	Warp of 32 threads
Memory	--	--	16 GB MCDRAM 384 GB DDR4	32 GB
Memory B/W	127.8 GB/s	127.8 GB/s	400+ GB/s (MCDRAM) 115.2 GB/s	900 GB/s
Approx. Unit Price	$1,200	$13,000	$3,400	$9,000

Demystifying the Cloud

- ”Regular” computers, just somewhere else
- Provide users with remote virtual machines or containers
- Can be used for anything:
  - Mobile-services, Hosting websites, Business Application, …
  - **Data Analysis, High Performance Computing**
- Providers
  - Major players: Amazon, Google, Microsoft, HP, IBM, Salesforce.com, …
  - Lots of others

Cloud Computing

- Advantages:
  - *Potentially* lower cost:
    - Pay as you go
  - *Potentially* lower cost:
    - Save on sysadmins and infrastructure
  - *Potentially* lower cost:
    - Economy of scale: providers
  - Scaling up or down as needed
    - Can be used to run overflow from a regular data center
  - Access to a wide range of hardware
- Additional challenges
  - Data movement
    - Expensive and time consuming
  - Security, privacy, …

What We Didn't Talk About

- Pipelining
- Microarchitecture (ALU, FPU, …)
- Cache lines and cache coherence
- Virtual memory
- Interconnects (network)
- FPGAs

Resources & References

- Very nice glossary: https://cvw.cac.cornell.edu/main/glossary
- J. Hennessy, D. Patterson, Computer Architecture: A Quantitative Approach, 6th edition (2017), ISBN 978-0128119051
- U. Drepper, What Every Programmer Should Know About Memory, http://people.redhat.com/drepper/cpumemory.pdf