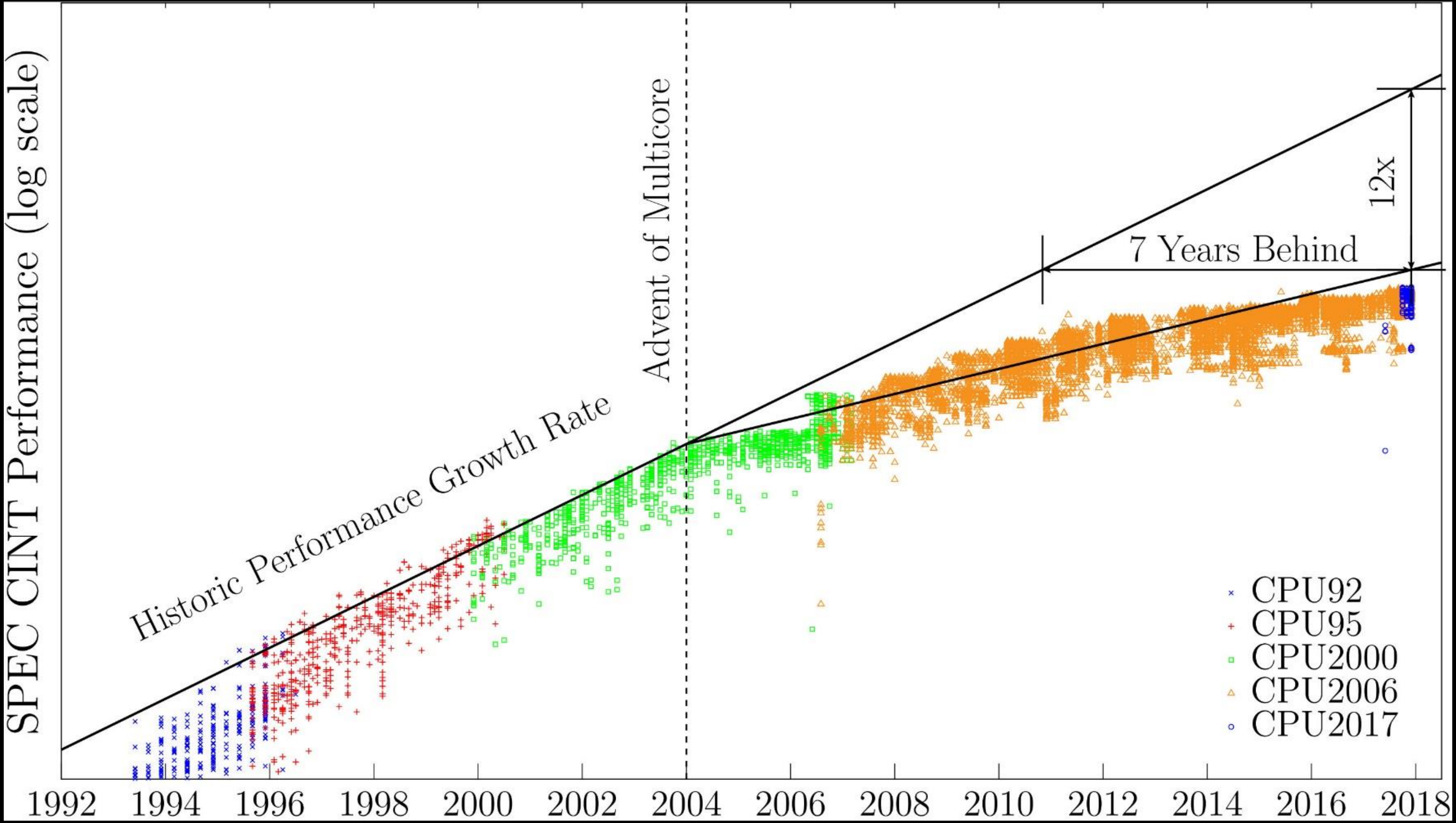
It’s a Multicore World

John Urbanic
Pittsburgh Supercomputing Center
Parallel Computing Scientist

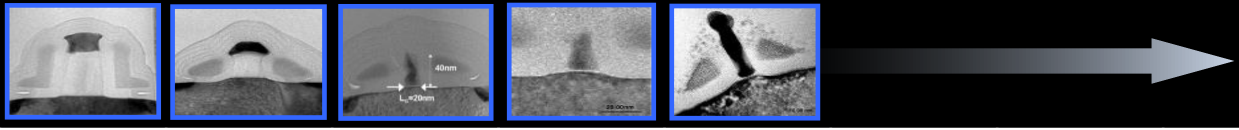
Moore's Law abandoned serial programming around 2004



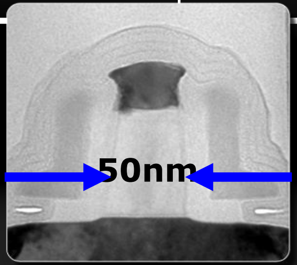
Courtesy Liberty Computer Architecture Research Group

Moore's Law is not to blame.

Intel process technology capabilities

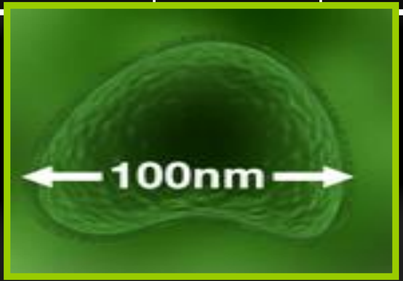


High Volume Manufacturing	2004	2006	2008	2010	2012	2014	2016	2018
Feature Size	90nm	65nm	45nm	32nm	22nm	16nm	11nm	8nm
Integration Capacity (Billions of Transistors)	2	4	8	16	32	64	128	256



Transistor for 90nm Process

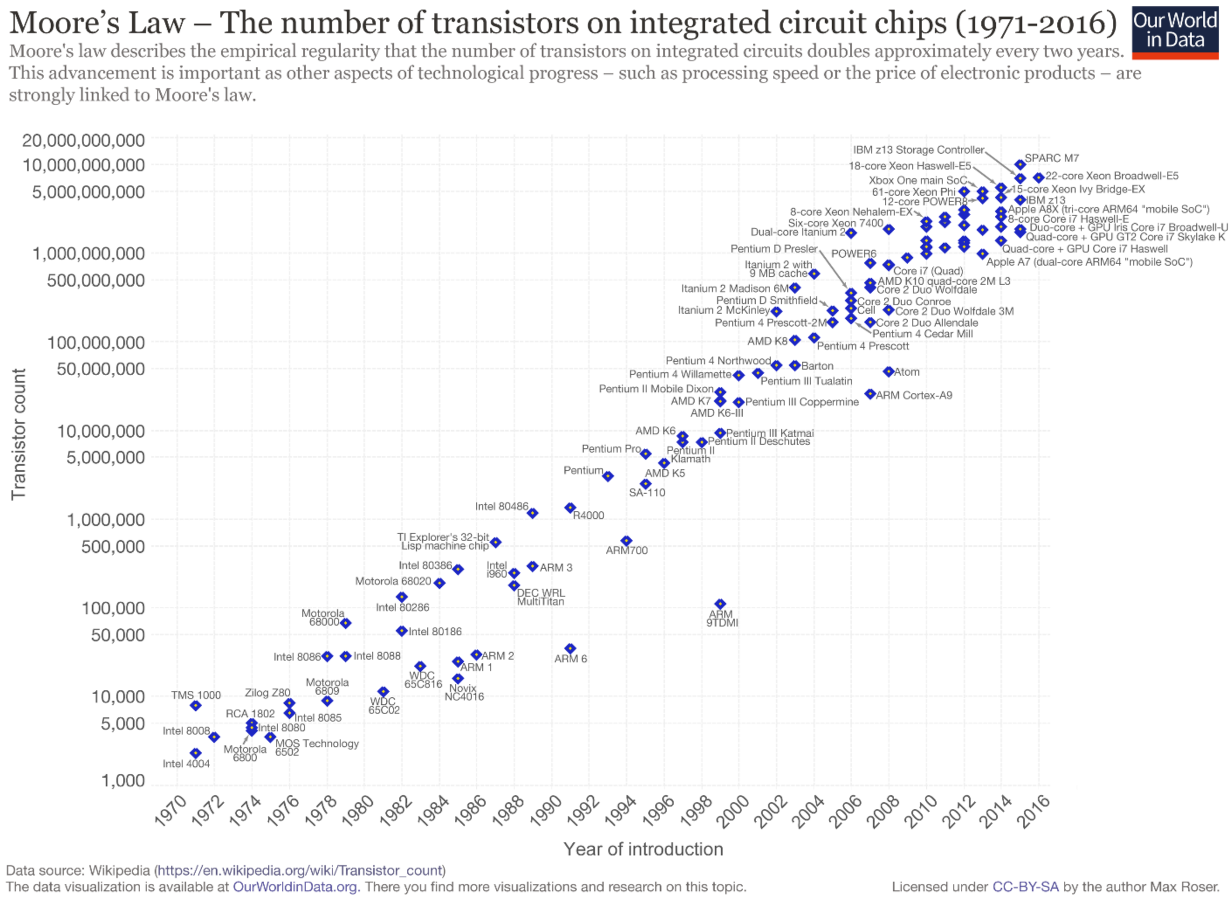
Source: Intel



Influenza Virus

Source: CDC

At end of day, we keep using all those new transistors.

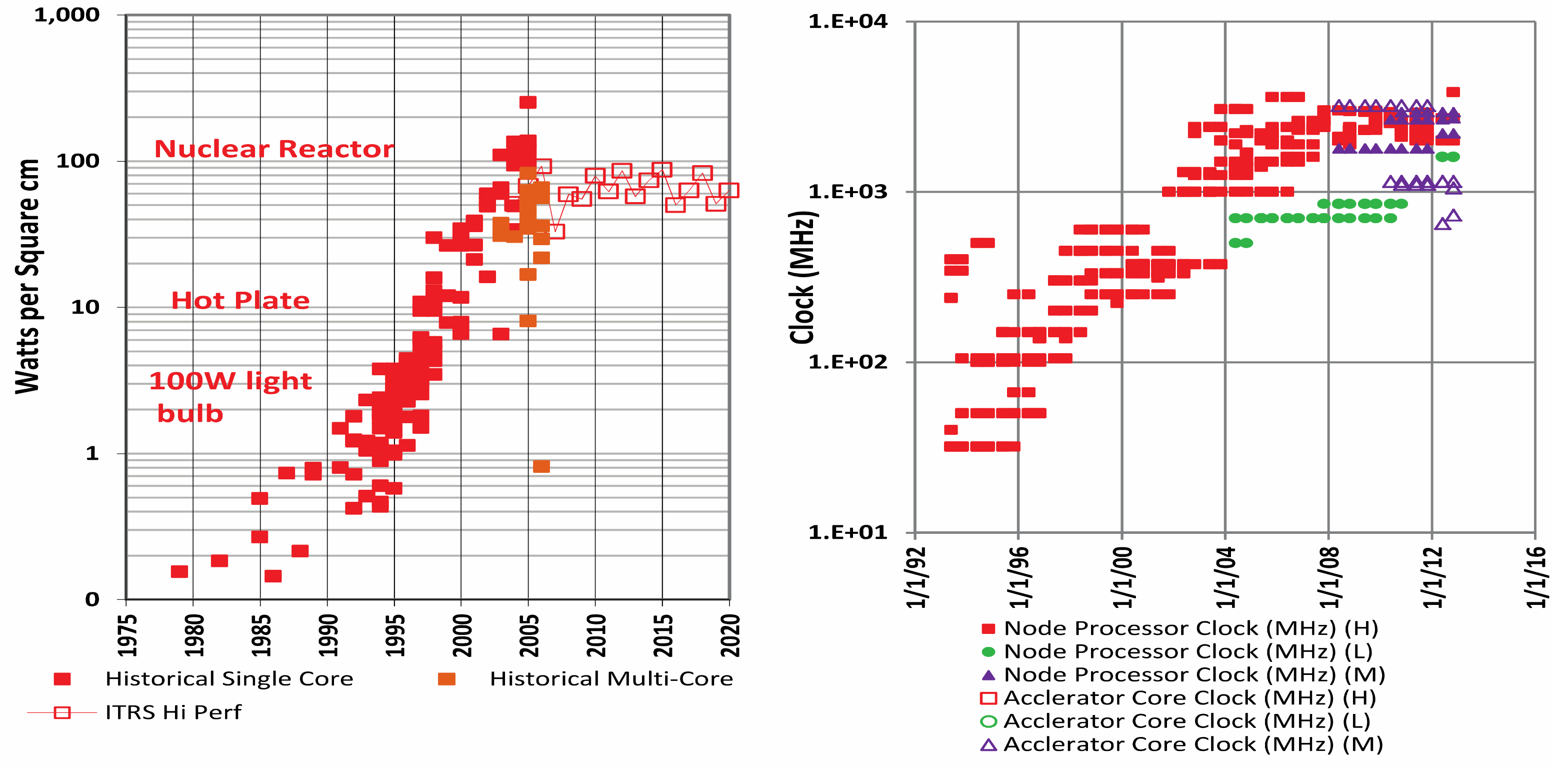


Moore's Law – The number of transistors on integrated circuit chips (1971-2016)

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.

Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)
The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

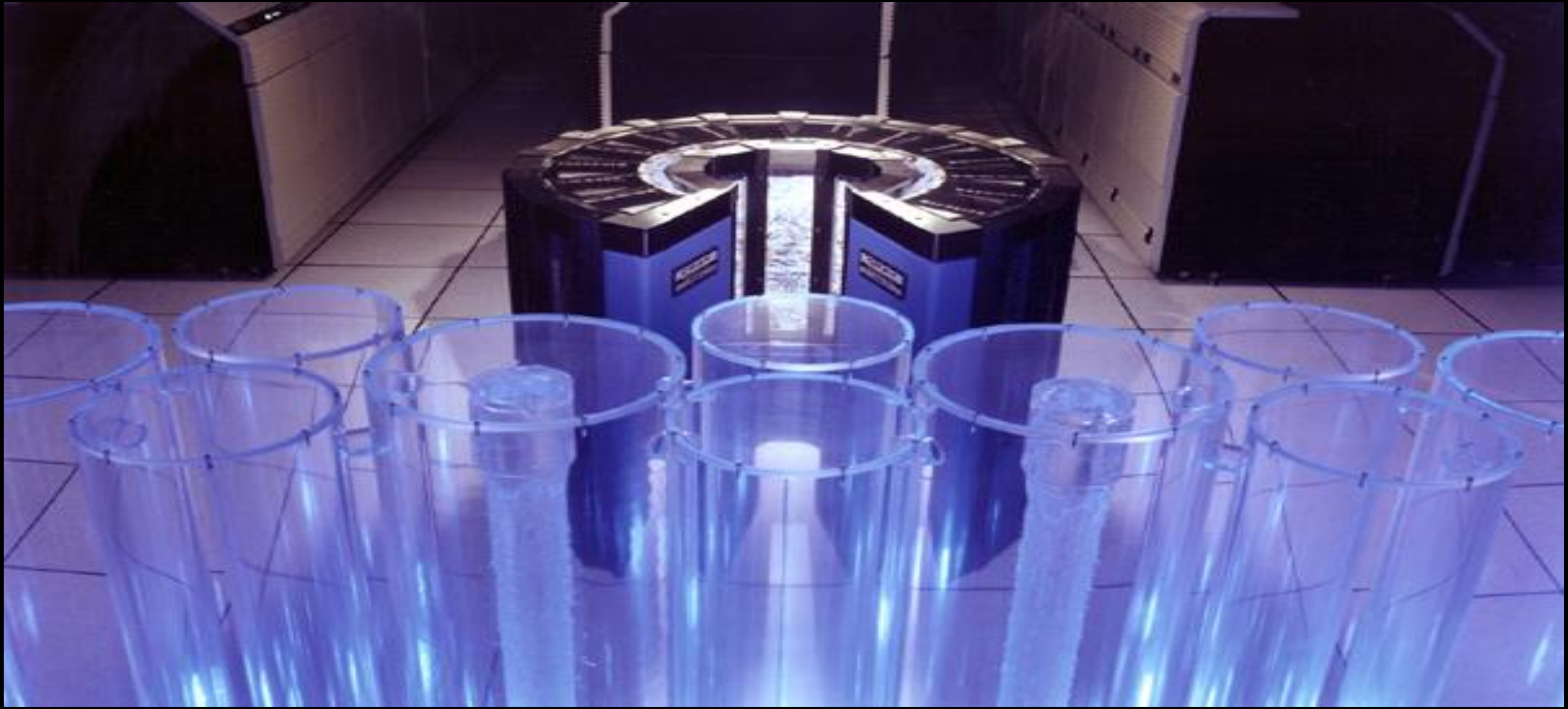
That Power and Clock Inflection Point in 2004… didn’t get better.



Fun fact: At 100+ Watts and <1V, currents are beginning to exceed 100A at the point of load!

Courtesy Horst Simon, LBNL

Not a new problem, just a new scale…



Cray-2 with cooling tower in foreground, circa 1985

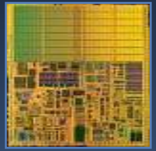
And how to get more performance from more transistors with the same power.

A 15% Reduction In Voltage Yields

RULE OF THUMB

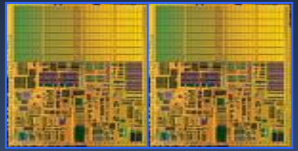
Frequency Reduction	Power Reduction	Performance Reduction
15%	45%	10%

SINGLE CORE



Area = 1
Voltage = 1
Freq = 1
Power = 1
Perf = 1

DUAL CORE

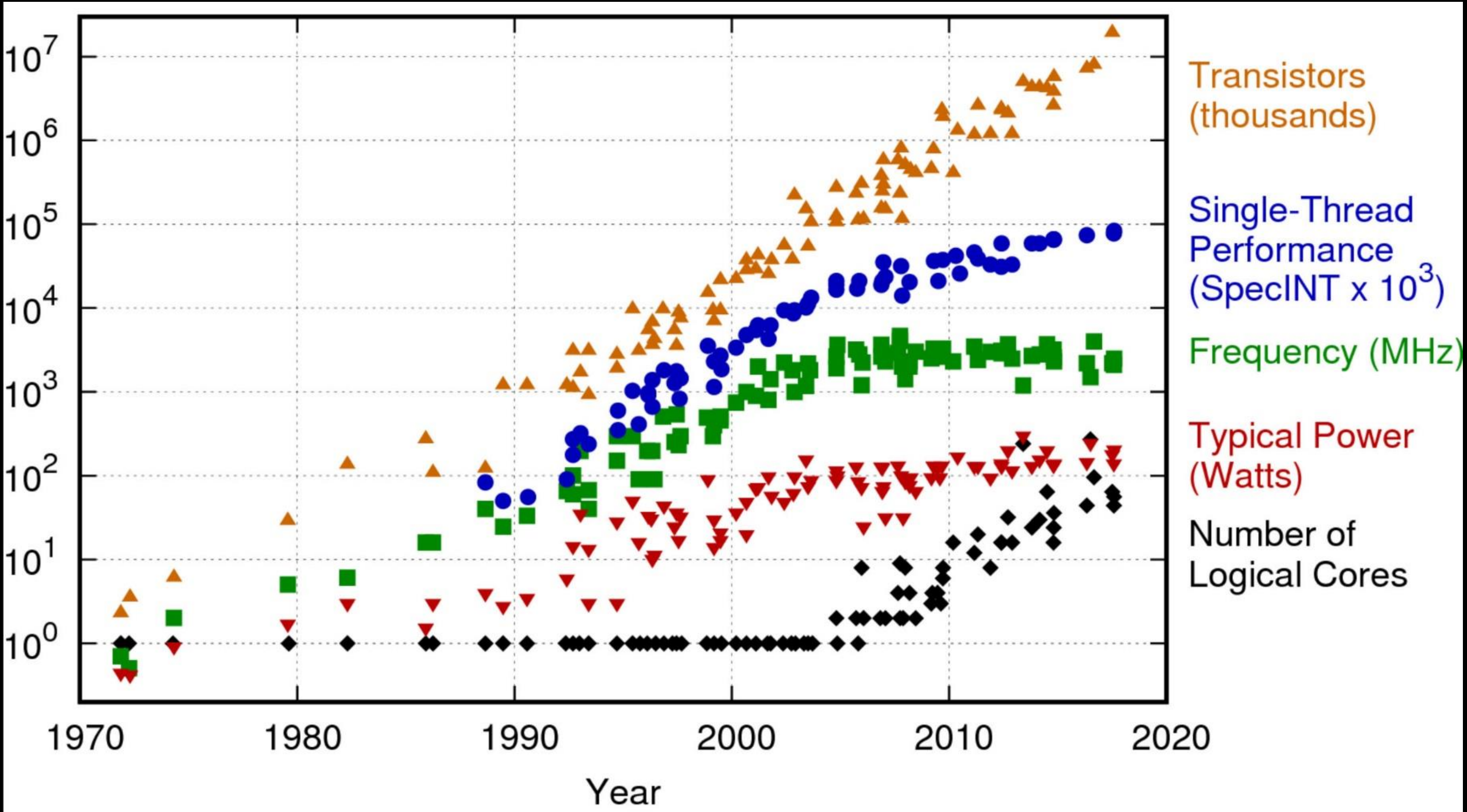


Area = 2
Voltage = 0.85
Freq = 0.85
Power = 1
Perf = ~1.8

Single Socket Parallelism

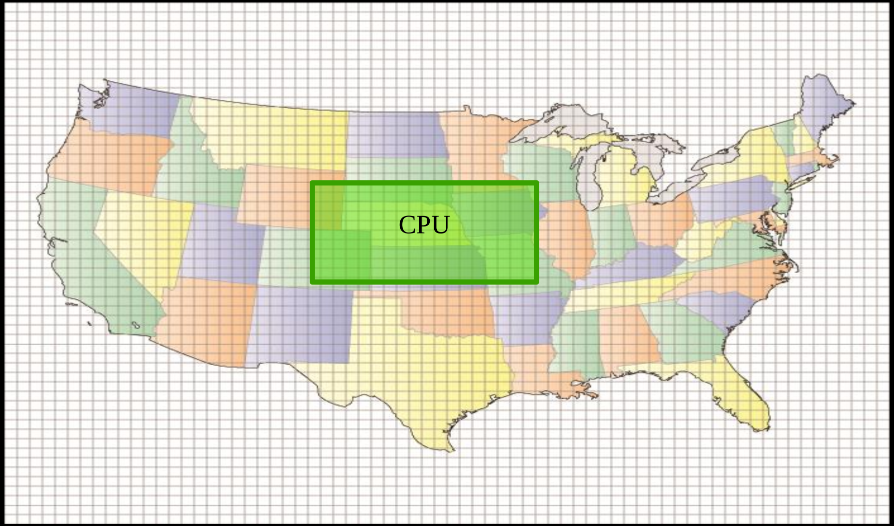
Processor	Year	Vector	Bits	SP FLOPs / core / cycle	Cores	FLOPs/cycle
Pentium III	1999	SSE	128	3	1	3
Pentium IV	2001	SSE2	128	4	1	4
Core	2006	SSE3	128	8	2	16
Nehalem	2008	SSE4	128	8	10	80
Sandybridge	2011	AVX	256	16	12	192
Haswell	2013	AVX2	256	32	18	576
KNC	2012	AVX512	512	32	64	2048
KNL	2016	AVX512	512	64	72	4608
Skylake	2017	AVX512	512	96	28	2688

Putting It All Together

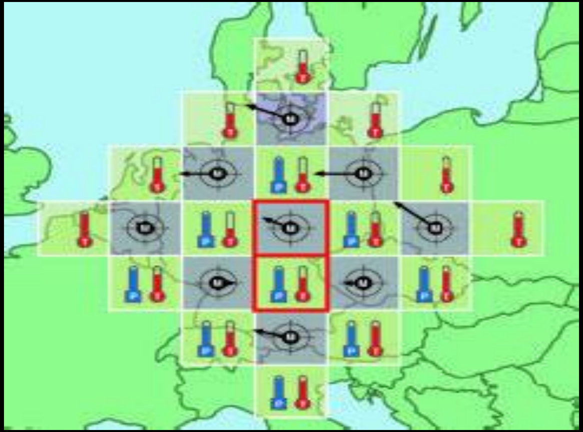


Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

Prototypical Application: Serial Weather Model

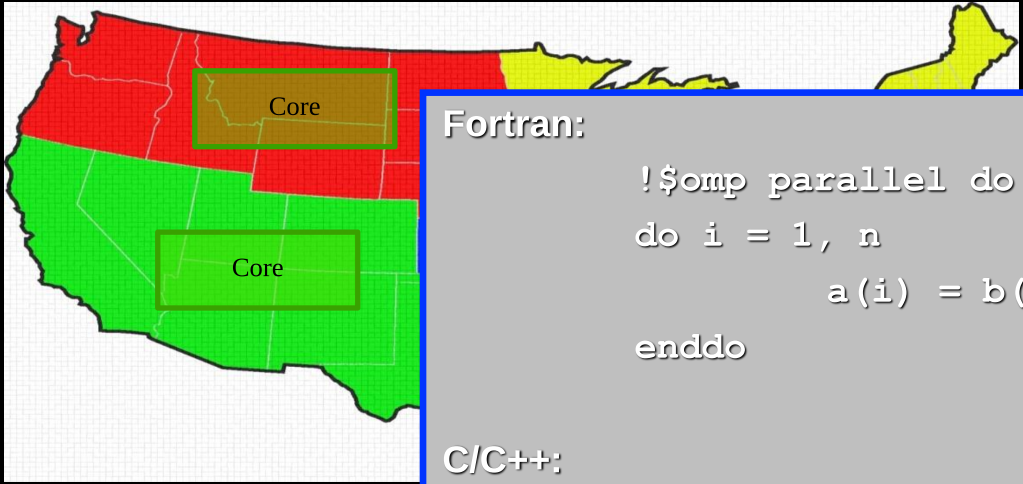


First Parallel Weather Modeling Algorithm: Richardson in 1917



Courtesy John Burkhardt, Virginia Tech

Weather Model: Shared Memory (OpenMP)



Fortran:

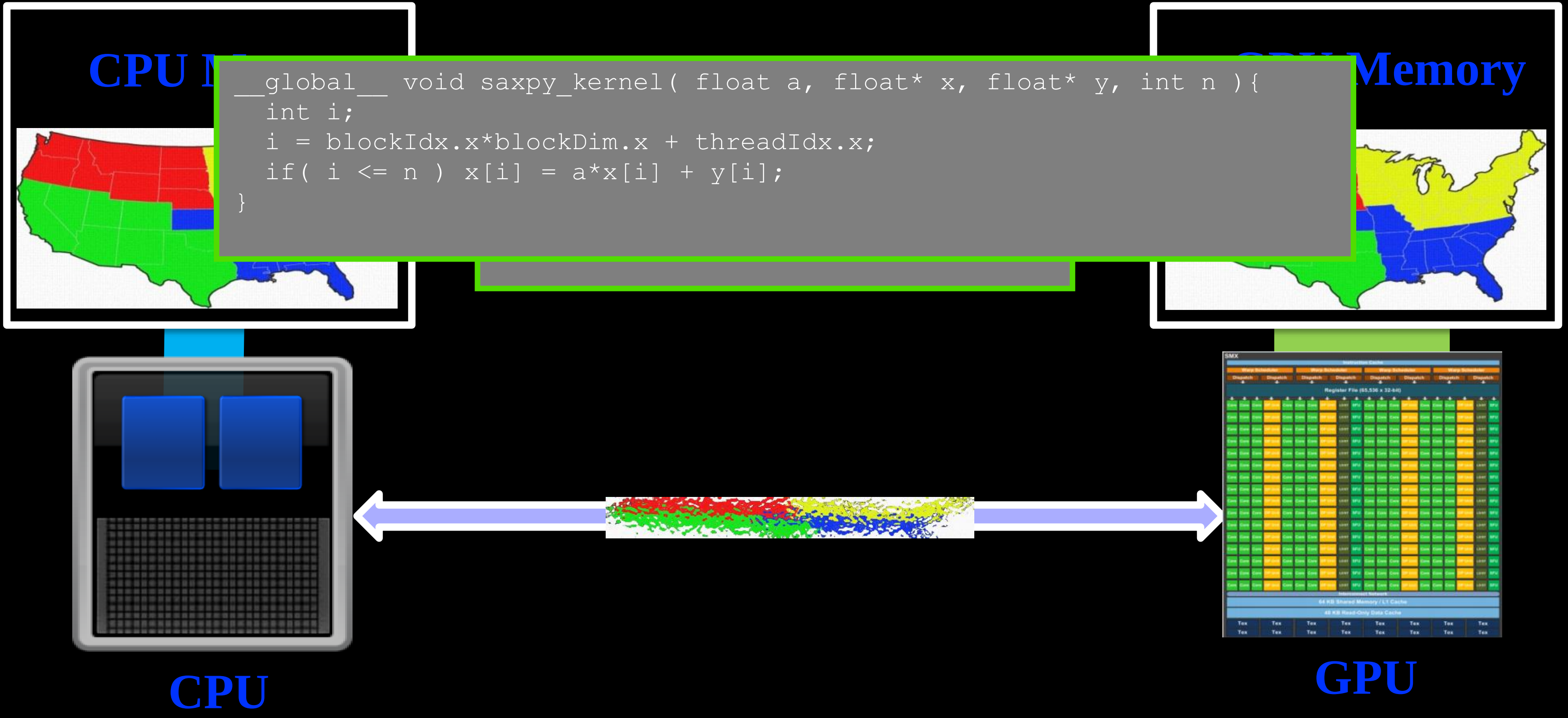
```
!$omp parallel do
do i = 1, n
        a(i) = b(i) + c(i)
enddo
```

C/C++:

```
#pragma omp parallel for
for(i=1; i<=n; i++)
          a[i] = b[i] + c[i];
```

Four meteorologists in th

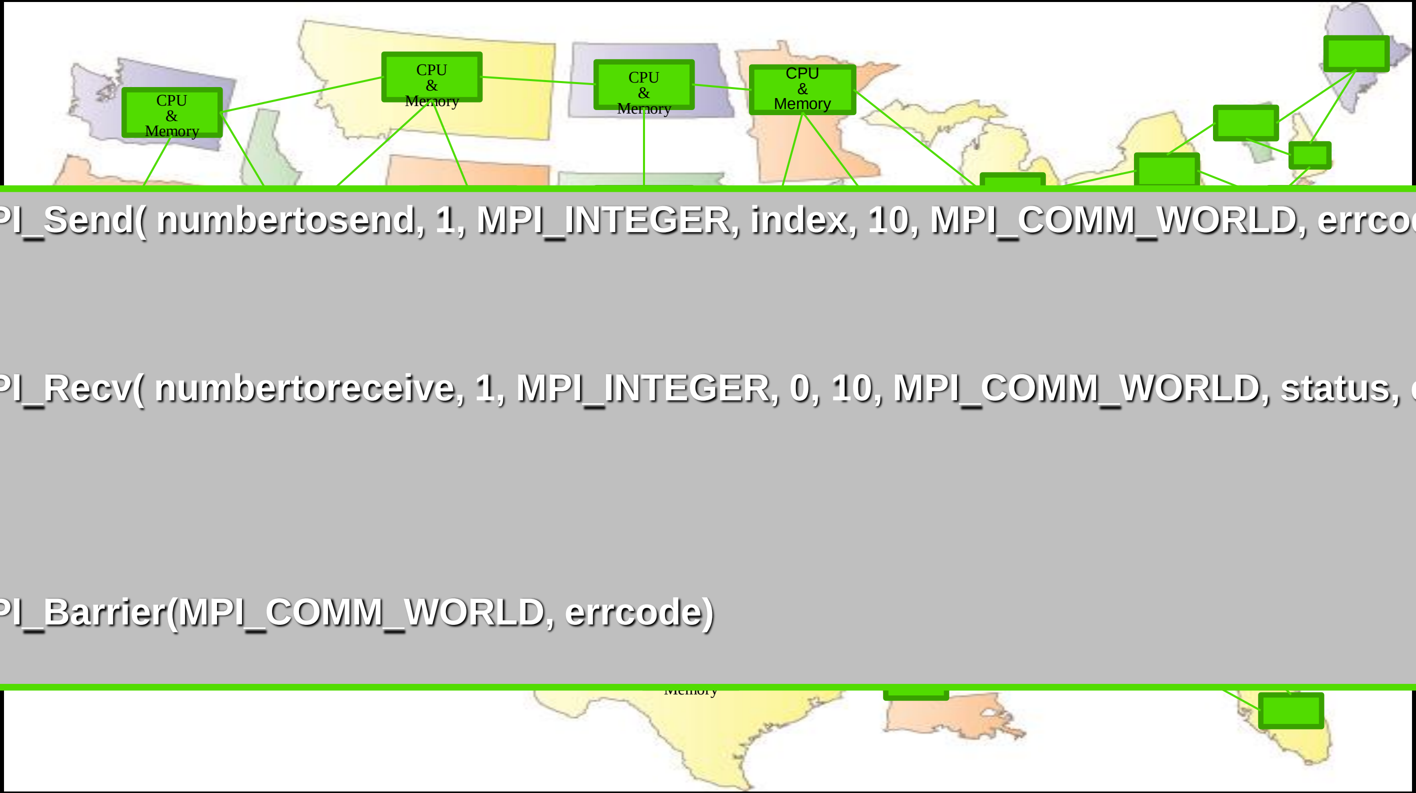
Weather Model: Accelerator (OpenACC)



```
__global__ void saxpy_kernel( float a, float* x, float* y, int n ){
  int i;
  i = blockIdx.x*blockDim.x + threadIdx.x;
  if( i <= n ) x[i] = a*x[i] + y[i];
}
```

1 meteorologists coordinating 1000 math savants using tin cans and a string.

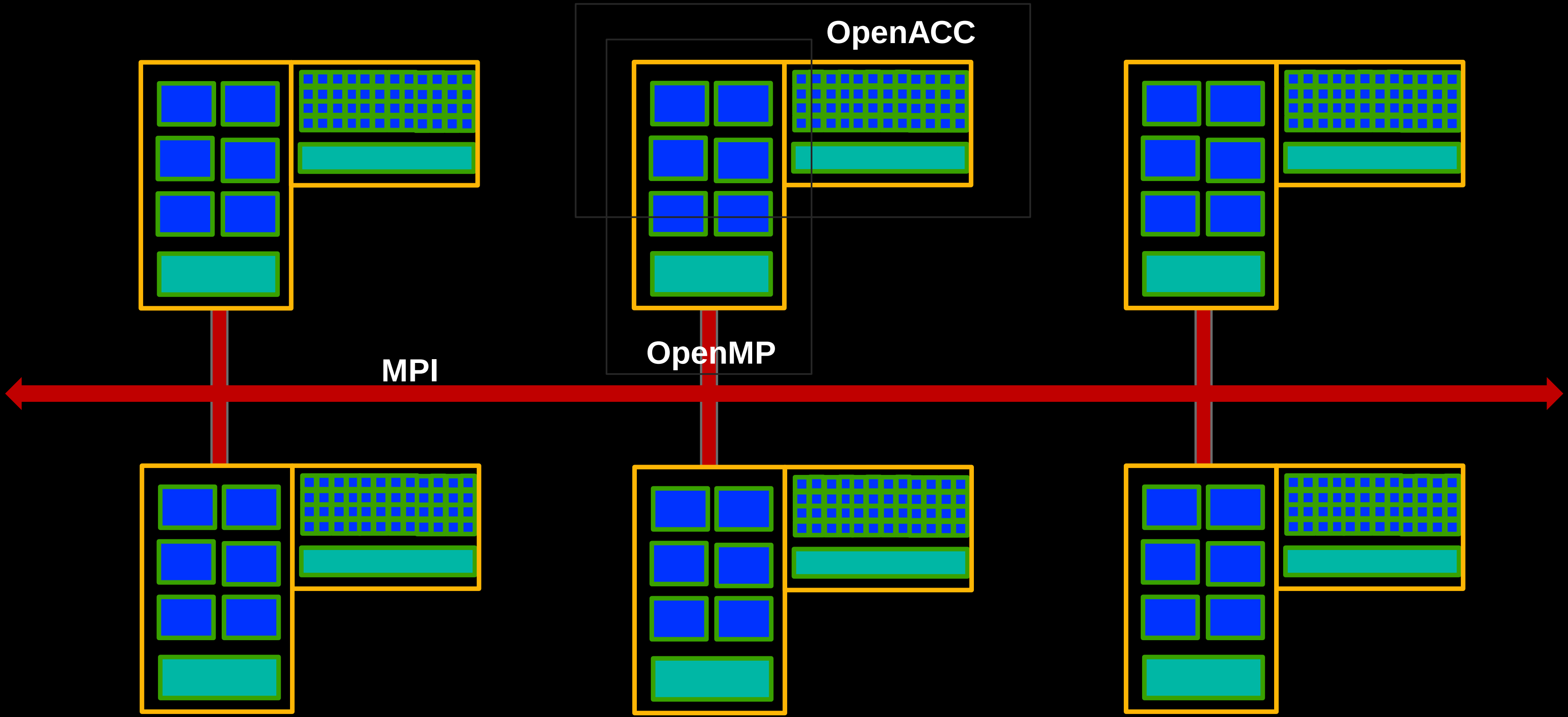
Weather Model: Distributed Memory (MPI)



```
call MPI_Send( numbertosend, 1, MPI_INTEGER, index, 10, MPI_COMM_WORLD, errcode)
.
.
call MPI_Recv( numbertoreceive, 1, MPI_INTEGER, 0, 10, MPI_COMM_WORLD, status, errcode)
.
.
.
call MPI_Barrier(MPI_COMM_WORLD, errcode)
.
```

50 meteorologists using a telegraph.

The pieces fit like this…



Many Levels and Types of Parallelism

- Vector (SIMD)
- Instruction Level (ILP)
  - Instruction pipelining
  - Superscaler (multiple instruction units)
  - Out-of-order
  - Register renaming
  - Speculative execution
  - Branch prediction

Compiler
(not your problem)

OpenMP

- Multi-Core (Threads)
- SMP/Multi-socket

OpenACC

- Accelerators: GPU & MIC

MPI

- Clusters
- MPPs

Also Important

- ASIC/FPGA/DSP
- RAID/IO

Cores, Nodes, Processors, PEs?

- The most unambiguous way to refer to the smallest useful computing device is as a Processing Element, or PE.

- This is usually the same as a single core.

- “Processors” usually have more than one core – as per the previous list.

- “Nodes” is commonly used to refer to an actual physical unit, most commonly a circuit board or blade with a network connection. These often have multiple processors.

I will try to use the term PE consistently here, but I may slip up myself. Get used to it as you will quite often hear all of the above terms used interchangeably where they shouldn’t be.

MPPs (Massively Parallel Processors)

Distributed memory at largest scale. Shared memory at lower level.

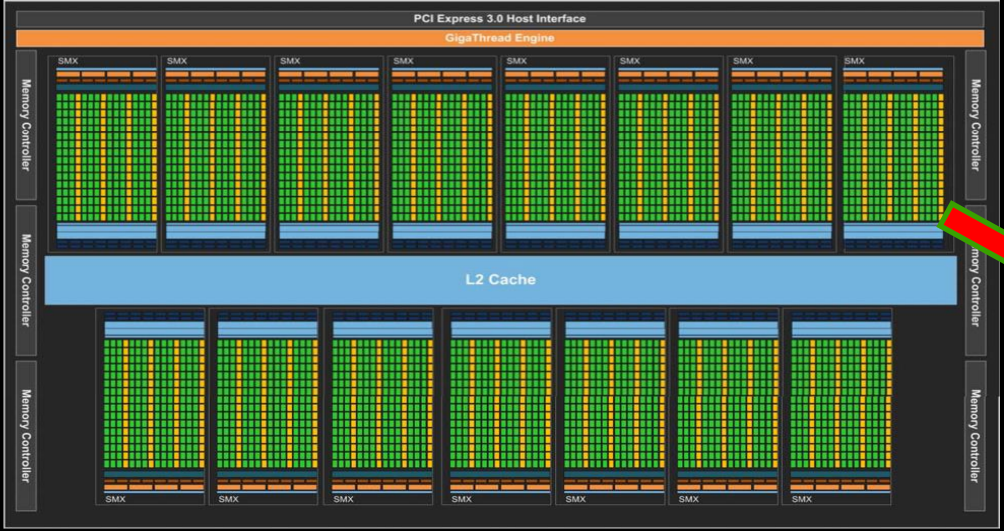
Summit (ORNL)

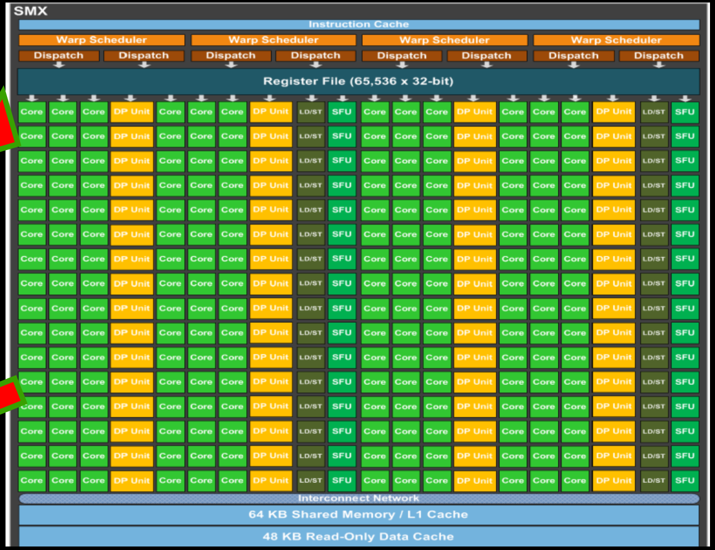
- 122 PFlops Rmax and 187 PFlops Rpeak
- IBM Power 9, 22 core, 3GHz CPUs
- 2,282,544 cores
- NVIDIA Volta GPUs
- EDR Infiniband

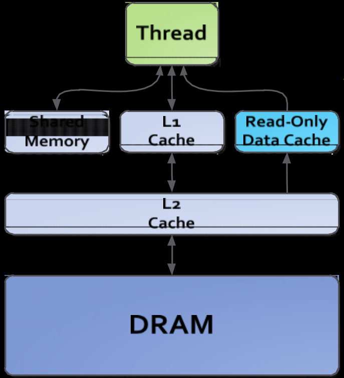
Sunway TaihuLight (NSC, China)

- 93 PFlops Rmax and 125 PFlops Rpeak
- Sunway SW26010 260 core, 1.45GHz CPU
- 10,649,600 cores
- Sunway interconnect

GPU Architecture - GK110 Kepler




SMX
Instruction Cache
Warp Scheduler
Dispatch
Register File (65,536 x 32-bit)
Core
DP Unit
LD/ST
SFU
Interconnect Network
64 KB Shared Memory / L1 Cache
48 KB Read-Only Data Cache



Thread

Shared Memory

L1 Cache

Read-Only Data Cache

L2 Cache

DRAM

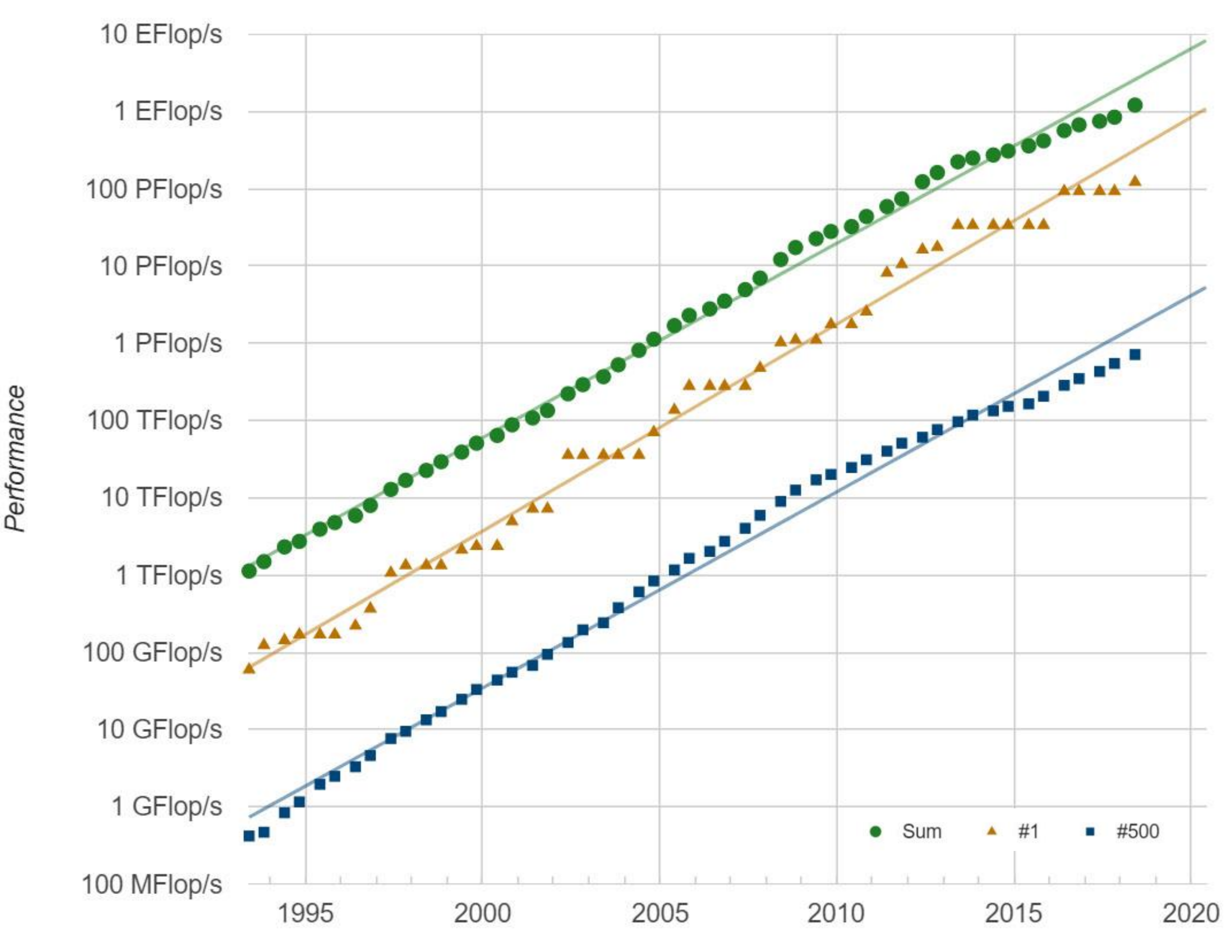
From a document you should read if you are interested in this:

http://www.nvidia.com/content/PDF/kepler/NVIDIA-Kepler-GK110-Architecture-Whitepaper.pdf

Top 10 Systems as of June 2020

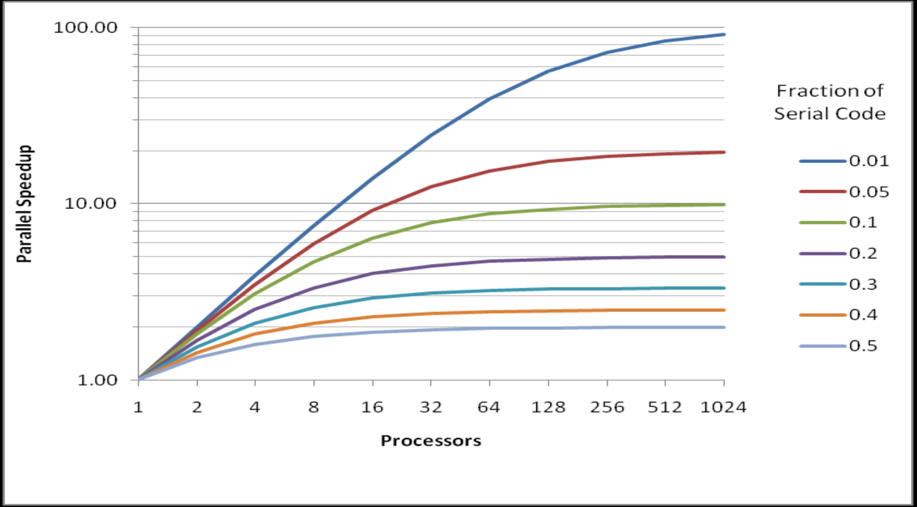
#	Site	Manufacturer	Computer	CPU Interconnect [Accelerator]	Cores	Rmax (Tflops)	Rpeak (Tflops)	Power (MW)
1	RIKEN Center for Computational Science **Japan**	Fujitsu	Fugaku	ARM 8.2A+ 48C 2.2GHz Torus Fusion Interconnect	**7,299,072**	**415,530**	**513,854**	**28.3**
2	DOE/SC/ORNL **United States**	IBM	Summit	Power9 22C 3.0 GHz Dual-rail Infiniband EDR NVIDIA V100	**2,414,592**	**148,600**	**200,794**	**10.1**
3	DOE/NNSA/LLNL **United States**	IBM	Sierra	Power9 3.1 GHz 22C Infiniband EDR NVIDIA V100	**1,572,480**	**94,640**	**125,712**	**7.4**
4	National Super Computer Center in Wuxi **China**	NRCPC	Sunway TaihuLight	Sunway SW26010 260C 1.45GHz	**10,649,600**	**93,014**	**125,435**	**15.3**
5	National Super Computer Center in Guangzhou **China**	NUDT	Tianhe-2 (MilkyWay-2)	Intel Xeon E5-2692 2.2 GHz TH Express-2 Intel Xeon Phi 31S1P	**4,981,760**	**61,444**	**100,678**	**18.4**
6	Eni S.p.A **Italy**	Dell	HPc5	Xeon 24C 2.1 GHz Infiniband HDR NVIDIA V100	**669,760**	**35,450**	**51,720**	**2.2**
7	Eni S.p.A **Italy**	NVIDIA	Selene	EPYC 64C 2.25GHz Infiniband HDR NVIDIA A100	**272,800**	**27,580**	**34,568**	**1.3**
8	Texas Advanced Computing Center/Univ. of Texas **United States**	Dell	Frontera	Intel Xeon 8280 28C 2.7 GHz InfiniBand HDR	**448,448**	**23,516**	**38,745**	
9	Cineca **Italy**	IBM	Marconi100	Power9 16C 3.0 GHz Infiniband EDR NVIDIA V100	**347,776**	**21,640**	**29,354**	**1.5**
10	Swiss National Supercomputing Centre (CSCS) **Switzerland**	Cray	Piz Daint Cray XC50	Xeon E5-2690 2.6 GHz Aries NVIDIA P100	**387,872**	**21,230**	**27,154**	**2.4**

Sustaining Performance Improvements



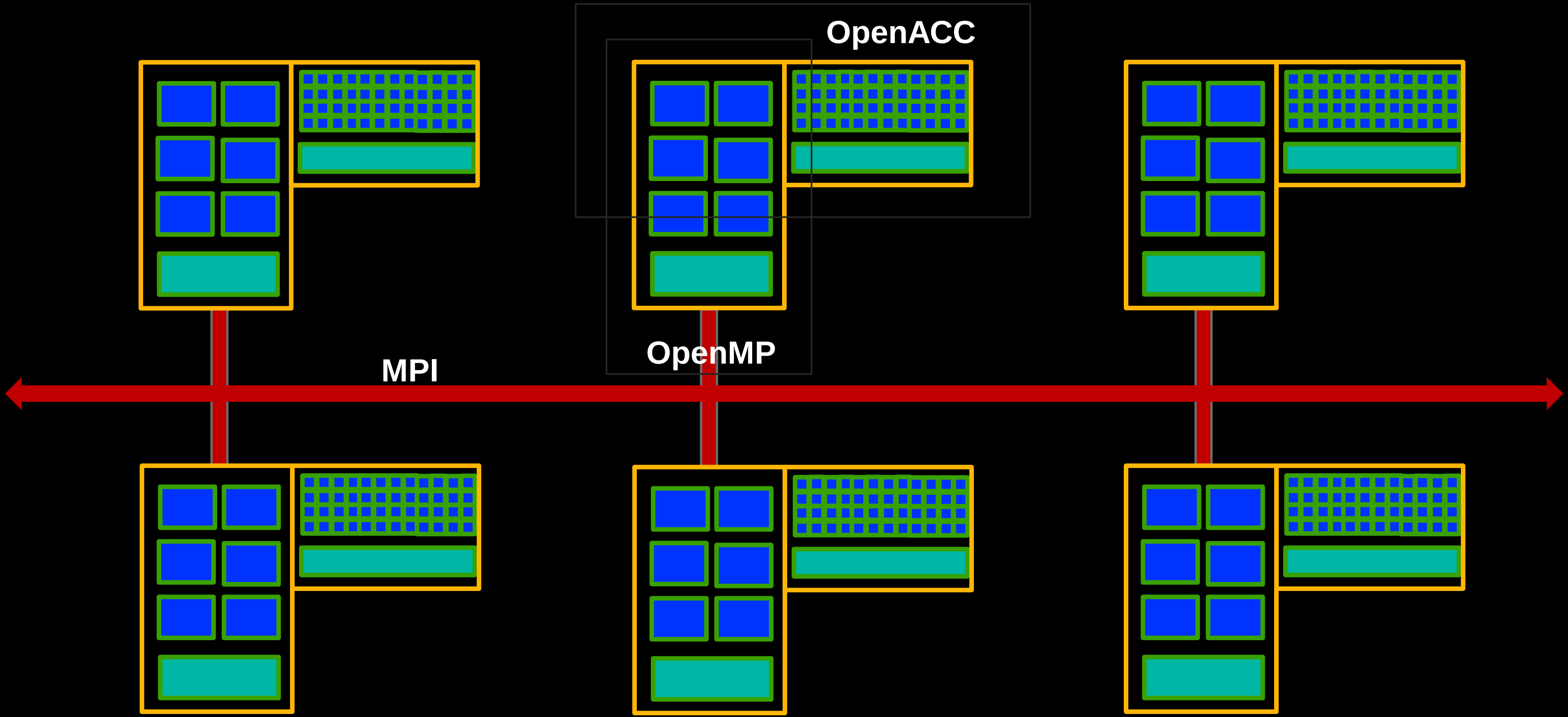
Amdahl's Law

- If there is x% of serial component, speedup cannot be better than 100/x.
- If you decompose a problem into many parts, then the parallel time cannot be less than the largest of the parts.
- If the critical path through a computation is T, you cannot complete in less time than T, no matter how many processors you use .



- Amdahl's law used to be cited by the knowledgeable as a limitation.
- These days it is mostly raised by the uninformed.
- Massive scaling is commonplace:
  - Science Literature
  - Web (map reduce everywhere)
  - Data Centers (Spark, etc.)
  - Machine Learning (GPUs and others)

In Conclusion…


OpenACC
OpenMP
MPI