Introduction to High-Performance Computing

Giorgio Amati Alessandro Ceci

Corso di dottorato in Ingegneria Aeronautica e Spaziale 2025 g.amati@cineca.it / g.amaticode@gmail.com alessandro.ceci@uniroma1.it

Agenda

- ✓ HPC: What is it?
- ✓ Hardware: how it works
 - Memory subsystem
 - Floating point units
- ✓ Algorithm vs. Implementation
- ✓ Compiler
- ✓ Parallel Paradigm
 - CPUs
 - GPUs
- ✓ Conclusions & Comments

Just for curiosity....

- ✓ Experience with HPC machine?
- ✓ Fortran, C, C++, anything else?
- ✓ Parallel paradigm: MPI, OpenMP, OpenACC, OpenMP offload,
- ✓ Linux, Windows, MacOS, (*NIX)
- ✓ Are you a Mathematician, a Physicist, an Engineer or a Computer Scientist?
- ✓ Do you know what is:
 - A Memory System?
 - A Cache?
 - A Floating Point Unit (FPU)?
 - A pipeline?
 - Moore Law?
 - Amdhal Law?

HPC: what it is?

From wikipedia:

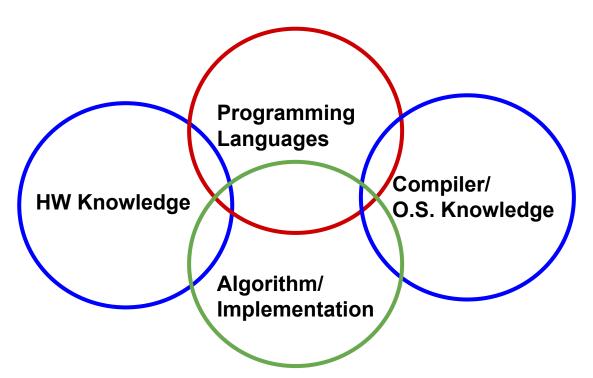
✓ "High-performance computing (HPC) uses supercomputers and computer clusters to solve advanced computation problems"

Personal definition:

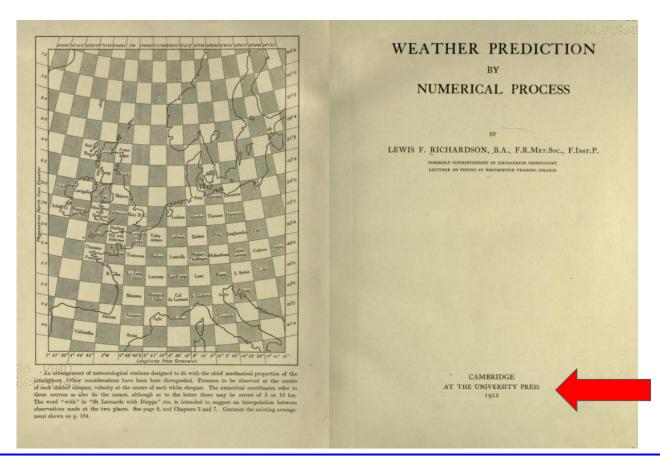
✓ It is the overlap of different skills, all devoted to exploit HW performance as much as possible (both serial and/or parallel, but not limited to supercomputers...)

HPC: what it is?

✓ These are the main skills for efficient HPC



HPC: older than computers?



HPC older than computers



Meteorologist Lewis Fry Richardson, creator of the first dynamic model for weather prediction, proposes the creation of a "forecast factory" that would employ some 64,000 human computers sitting in tiers around the circumference of a giant globe. Each calculator would be responsible for solving differential equations related to the weather in his quadrant of the earth. From a pedestal in the center of the factory, a conductor would orchestrate this symphony of equations by shining a beam of light on areas of the globe where calculation was moving too fast or falling behind.

https://www.historyofinformation.com/detail.php?id=59

Example: matrix-matrix multiplication

Simple problem: for 2 n² matrices we have to:

- ✓ compute n^3 products and n^3 sums
- ✓ load 2*n^2 data and to store n^2 data
 - Ratio computation vs. load/store is O(n)!

```
do j = 1, n
    do k = 1, n
        do i = 1, n
            c(i,j) = c(i,j) + a(i,k)*b(k,j)
        enddo
    enddo
enddo
```

✓ MM multiplication is used for supercomputing rankings (top500)

Example: matrix-matrix multiplication/2

- ✓ Performance can be really different, depending on HW, implementation etc....
- ✓ Performance improvements can be really high....
- ✓or you can easily "depress" performance
- ✓ Performance is measured in Mflops: the higher the better

#test	Size	HW	MFlops	Ratio
1	2048	1	201	-
2	2048	1	4870	24x
3	8192	2	361328	1797x
4	8192	2	448923	2233x

Few "facts" about HPC

HPC market is not big enough to survive...

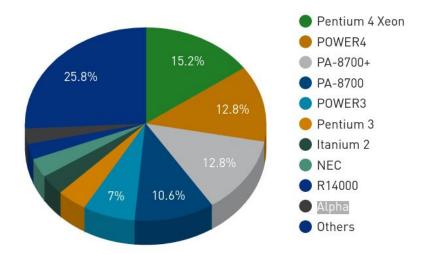
- ✓ SGI
- ✓ Compaq
- ✓ <u>Digital</u>
- ✓ SUN
- ✓ SiCortex
- ✓ MTA
- ✓ CRAY
- ✓ CONVEX
- ✓ CDC
- ✓ Thinking Machine
- ✓ Quadrics/APE

- ✓ IBM
 - Power3/4/.../9
- ✓ Intel
 - o <u>Itanium</u>
 - o Phi
- ✓ HP
- ✓ NVIDIA
- **✓ FUJITSU**
- ✓ AMD
 - Opteron
- ✓ NEC
 - <u>SX6</u>

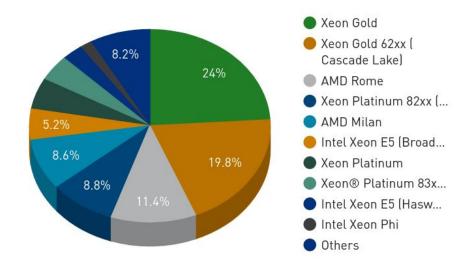
Few "facts" about HPC

Top500 list: <u>June 2003</u> vs <u>November 2022</u>

Processor Generation System Share



Processor Generation System Share



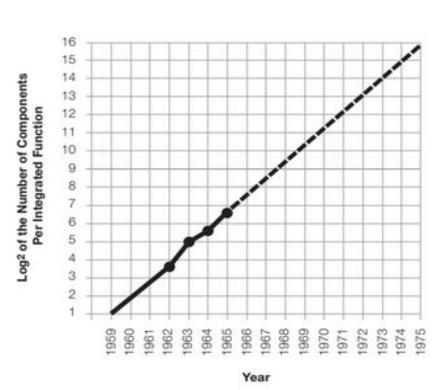
Moore's Law

In his 1965 article, Moore (Intel co-founder) planned the increase of the # of transistors up to 1975.

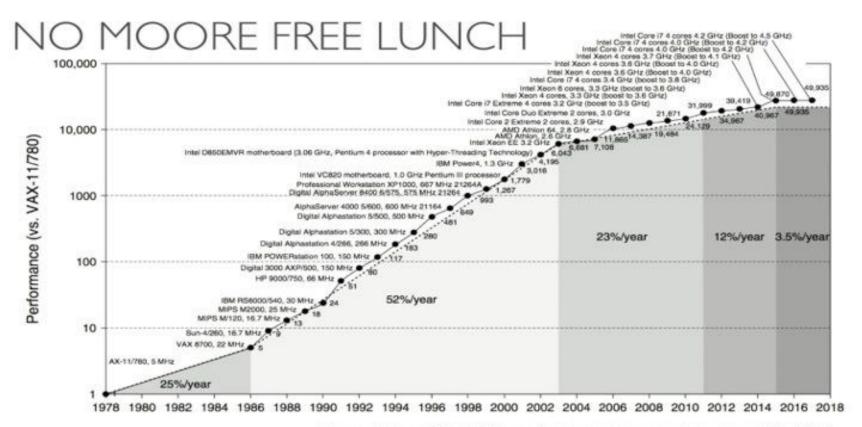
"With unit cost falling as the number of components per circuit rises, by 1975 economics may dictate squeezing as many as 65,000 components on a single silicon chip"

He stated a 2x increment of transistors every 18 Month.

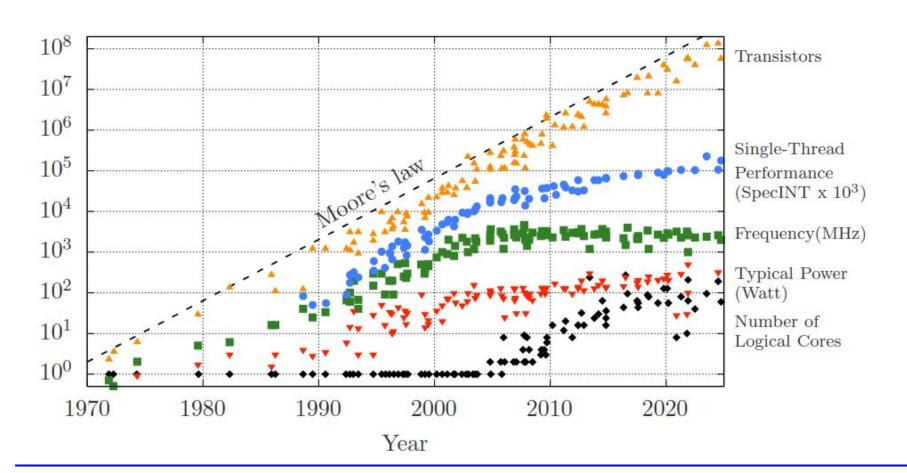
- This law is still valid now (in some form): to "survive" in the market HW firms must follow this law
- Now "transistor shrinking" is much harder: we are near to quantum effects
- ✓ New "ideas" must be found



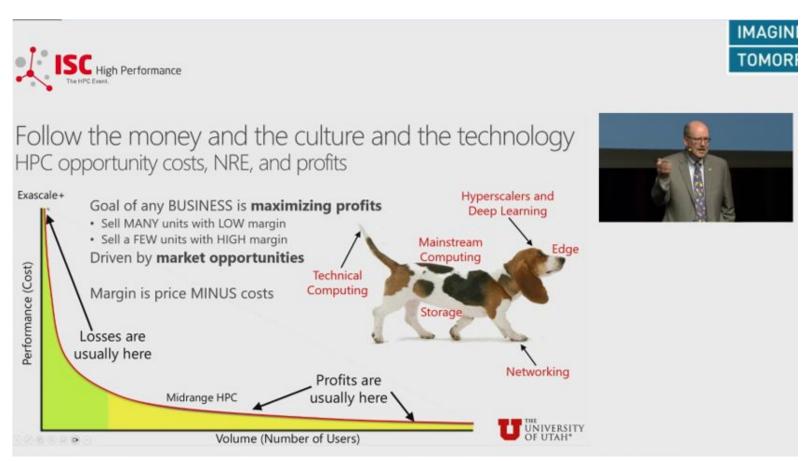
Moore's Law



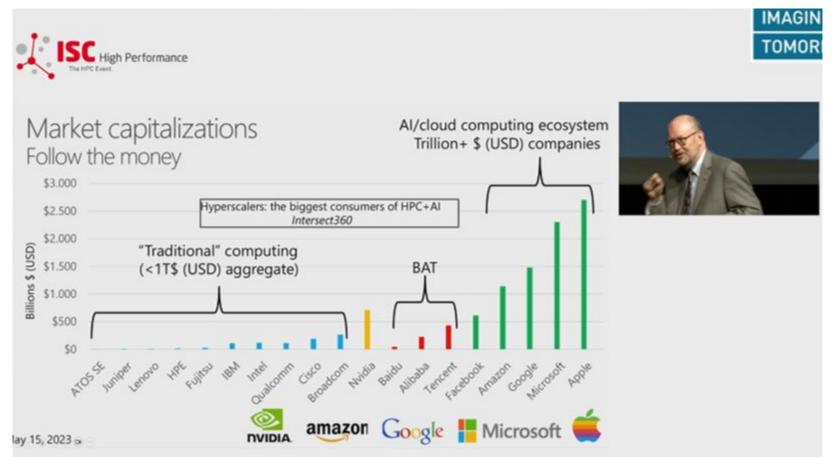
Moore's Law



Follow the money! (from D.Reed@ISC2023)



Follow the money! (from D.Reed@ISC2023)



Follow the money! (from D.Reed@ISC2023)



IMAGINE TOMORROW

Al, cloud, and silicon innovation Follow the money and the culture ...

Hardware unicorns and AI startups

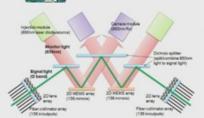
- · Cerebras, GraphCore, Grog, Hailo
- · SambaNova, Wave Computing, ...

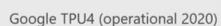












- 4096 units per "pod" (1.1 exaops)
- 3-D twisted torus optical interconnect

AWS Graviton3

- · 64 ARM Neoverse V1 cores, 7 chiplet design
- · 55 billion transistors, DDR5 memory, PCle5

Microsoft Azure (XCG legacy, in part)

- · Ampere ARM and Project Catapult/Brainwave
- · \$10B+ OpenAl investment

Ampere One (192 cores, Bfloat16)





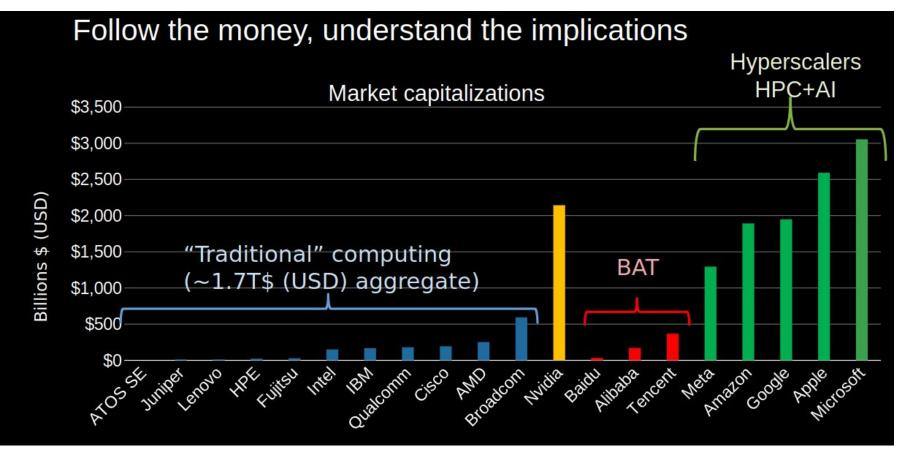
Google







Follow the money!/2 (from Yelick@ISC2024)



For Boomer only: Palm pilot (PDA)

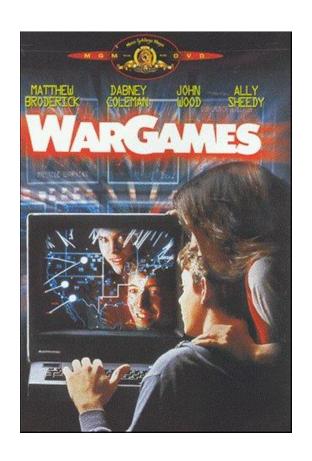
Do we remember there was a company called Palm, which used to be the pioneer in Personal Digital Assistant (PDA) industry, they reached the peak of success in 2000 with a market cap exceeding Apple, Amazon, Google, and Nvidia combined.

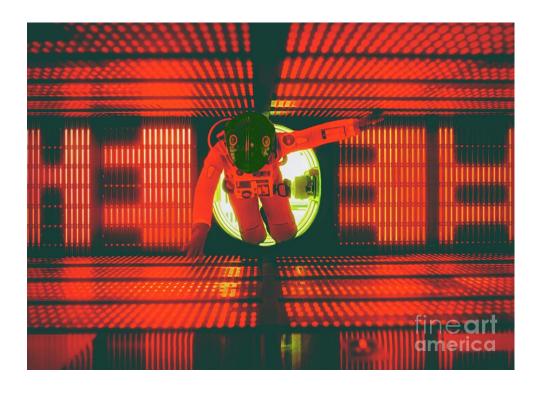
However, their attachment with physical buttons and slowness to adapt to touchscreens led to their downfall. HP acquired palm in 2009 and tried to revive the industry but failed and was discontinued in 2010.

Moral: Today's tech titan could be tomorrow's trivia answer. In this brutal tech world, any tech startup is as good as their last update.



HPC@movie





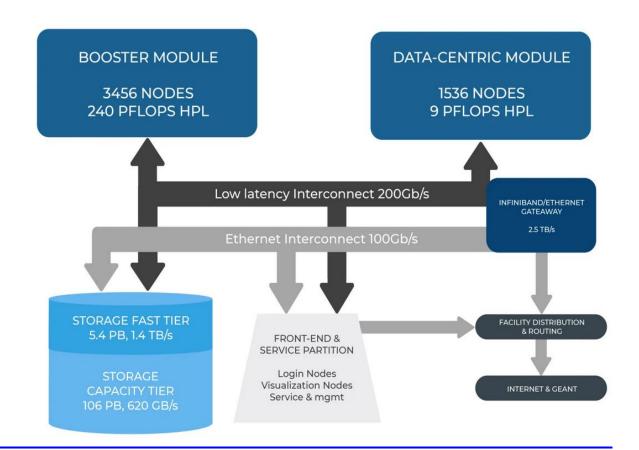


Leonardo

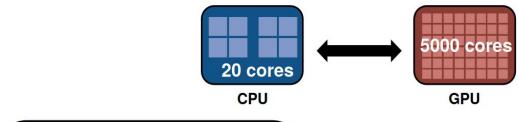


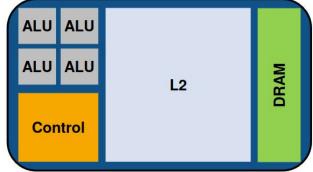
Leonardo: main figures

- ✓ 1536 CPU-based nodes
 - 172032 cores
- √ 3456 GPU-based nodes
 - 13824 GPU
 - 110592 cores
- ✓ 155 Racks
 - 16 CPU racks
 - 116 GPU racks
 - 12 I/O racks
 - 1 System racks
 - About 300'000 Kg!
- ✓ Power Requirements
 - HPL: ~ 8.0 MW
 - Operational: ~ 6.0 MW



CPU vs. GPU





- ✓ Optimized for low latencies
- ✓ Huge caches
- ✓ Control logic for out-of-order and speculative execution
- ✓ Targets on general-purpose applications



- ✓ Optimized for data parallel throughput
- ✓ Memory latency tolerant
- ✓ More transistors dedicated to computations
- ✓ Targets on special applications

Why GPUs?

✓ Pro

- GPUs are more powerful: 1 GPU ~ 10x CPUs (Peak Mflops)
- GPUs ask for less space: for same performance CPUs ask for ~3x racks
- GPUs are less expensive: for same peak performance CPUs are ~2x expensive
- GPUs ask for (relative) less power: for same peak performance CPUs ask ~4x energy

✓ Cons

- GPUs are less flexible respect CPUs
- Some algorithm are not GPU-friendly
- There's no a common programming model between different vendors
- Porting to GPU is expensive and error-prone procedure

Rank	System	Cores	Rmax (GFlop/s)	Rpeak (GFlop/s)	Power (kW)
1	Earth-Simulator, NEC Japan Agency for Marine-Earth Science and Technology Japan	5,120	35,860.00	40,960.00	3,200
2	ASCI Q - AlphaServer SC45, 1.25 GHz, HPE Los Alamos National Laboratory United States	8,192	13,880.00	20,480.00	
3	X - 1100 Dual 2.0 GHz Apple G5/Mellanox Infiniband 4X/Cisco GigE, Self-made Virginia Tech United States	2,200	10,280.00	17,600.00	
4	Tungsten - PowerEdge 1750, P4 Xeon 3.06 GHz, Myrinet, DELL EMC NCSA United States	2,500	9,819.00	15,300.00	
5	Mpp2 - Cluster Platform 6000 rx2600 Itanium2 1.5 GHz, Quadrics, HPE DOE/SC/Pacific Northwest National Laboratory United States	1,936	8,633.00	11,616.00	
6	Lightning - Opteron 2 GHz, Myrinet, Linux Networx Los Alamos National Laboratory United States	2,816	8,051.00	11,264.00	

Top500 (20 years ago)

- 1 Self-made Supercomputer
- No GPU
- 1 Vector Machine (NEC)
- 5 different vendor/integrator
- 3.2 MW for #1
- ~10 TF/MW
- Linux Network
- **-NEC**

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	Tianhe-2A - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P, NUDT National Super Computer Center in Guangzhou China	3,120,000	33,862.70	54,902.40	17,808
2	Titan - Cray XK7, Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x, Cray/HPE D0E/SC/Oak Ridge National Laboratory United States	560,640	17,590.00	27,112.55	8,209
3	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom, IBM DOE/NNSA/LLNL United States	1,572,864	17,173.22	20,132.66	7,890
4	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect, Fujitsu RIKEN Advanced Institute for Computational Science (AICS) Japan	705,024	10,510.00	11,280.38	12,660
5	Mira - BlueGene/Q, Power BQC 16C 1.60GHz, Custom, IBM DOE/SC/Argonne National Laboratory United States	786,432	8,586.61	10,066.33	3,945
6	Piz Daint - Cray XC30, Xeon E5-2670 8C 2.600GHz, Aries interconnect, NVIDIA K20x, Cray/HPE Swiss National Supercomputing Centre [CSCS] Switzerland	115,984	6,271.00	7,788.85	1,754

Top500 (10 years ago)

- 2 GPUs based
- 1 CPU based (K-Computer)
- 3 Manycore (BG/intel Phi)
- 4 different vendor/integrator
- Up to 30 MW
- ~1.9 PF/MW

← NUDT

IBM

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE D0E/SC/Oak Ridge National Laboratory United States	8,699,904	1,194.00	1,679.82	22,703
2	Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	4,742,808	585.34	1,059.33	24,687
3	Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Microsoft Azure United States	1,123,200	561.20	846.84	
4	Supercomputer Fugaku - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
5	LUMI - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	2,752,704	379.70	531.51	7,107
6	Leonardo - BullSequana XH2000, Xeon Platinum 8358 32C 2.6GHz, NVIDIA A100 SXM4 64 GB, Quad-rail NVIDIA HDR100 Infiniband, EVIDEN EuroHPC/CINECA Italy	1,824,768	238.70	304.47	7,404

Top500 (23/11)

- 5 GPU-based
- 1 CPU-based (Fugaku)
- 5 different vendor/integrator
- Up to 30 MW for #1
- ~43 PF/MW

Questions!

- #1: Why all this complexity?
- #2: Is the market big enough to survive for a HPC firm?
- #3: Which skill is the more important?
- #4: What is the performance range?
- #5: Why GPUs?
- #6: And for the next 20 years?

Recap

- ✓ Different skills are required to achieve "good" performance
- ✓ Performance is not only a problem of choosing the right (powerful) HW
- ✓ HW evolution is driven by mass market
- ✓ All firms devoted only to HPC have not survived to the market
- ✓ Users should (or must) be flexible enough to follow HW & SW evolution
- ✓ A Correct code could be efficient or not. With different order of magnitude!
- ✓ Today any processor is a parallel one
 - To have a parallel code doesn't mean to have an efficient one
- ✓ To be fast is secondary respect to be correct.
 - "Premature optimization is the root of all evils" (D. Knuth)
- ✓ But you'll must face optimization issues soon
 - 1 way to go fast, 100 ways to go slow!
- ✓ Today CPUs/GPUs can have order of 100'000'000'000 transistors

Take home message

- ✓ HPC is complicate stuff, many skills are needed
 - you have to know how HW works
 - you have to know how SW works
 - you have to know about numerics
 - you have to know about the problem to solve
- ✓ Good/Bad performances depends (also) from the user
- ✓ Things change over the year
- ✓ Sorry: no "silver bullet" or "free lunch"
- ✓ No one will develop a CPU/GPU specifically for you.....