Introduction to High-Performance Computing

Giorgio Amati

Corso di dottorato in Ingegneria Aeronautica e Spaziale 2024

g.amati@cineca.it / g.amaticode@gmail.com

CAVEAT

- ✓ This is only an introduction
- ✓ There’s much more under the hood
- ✓ These 6 lessons can only help you to understand some basic concept
- ✓ Needs more experience (and so many mistakes) to manage HPC
- ✓ I’m old guy: almost all examples are in Fortran but the reasons behind performance are (almost) language independent
- ✓ write to g.amaticode@gmail.com to have your e-mail
- ✓ material downloadable from: https://github.com/gamati01/HPCLessons

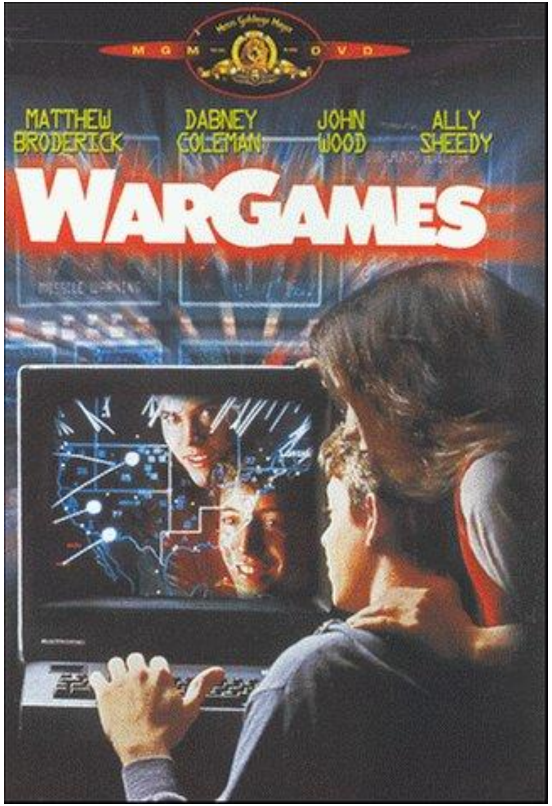
Agenda

- ✓ **HPC: What it is?**
- ✓ **Spoiler…**
- ✓ Hardware: how it works
- ✓ Algorithm vs. Implementation
- ✓ Compiler
- ✓ Parallel Paradigm
- ✓ Conclusions & Comments

Just for curiosity....

- ✓ Experience of HPC machine?
- ✓ Fortran, C, C++, everything else?
- ✓ Parallel paradigm: MPI, OpenMP, OpenACC, OpenMP offload, ….
- ✓ Linux, Windows, MacOS, (*NIX)
- ✓ Are you a Mathematician, a Physicist, a 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@movie


MGM DVD
MATTHEW BRODERICK
DABNEY COLEMAN
JOHN WOOD
ALLY SHEEDY
WARGAMES


fineart america

HPC: what it is?

From wikipedia:

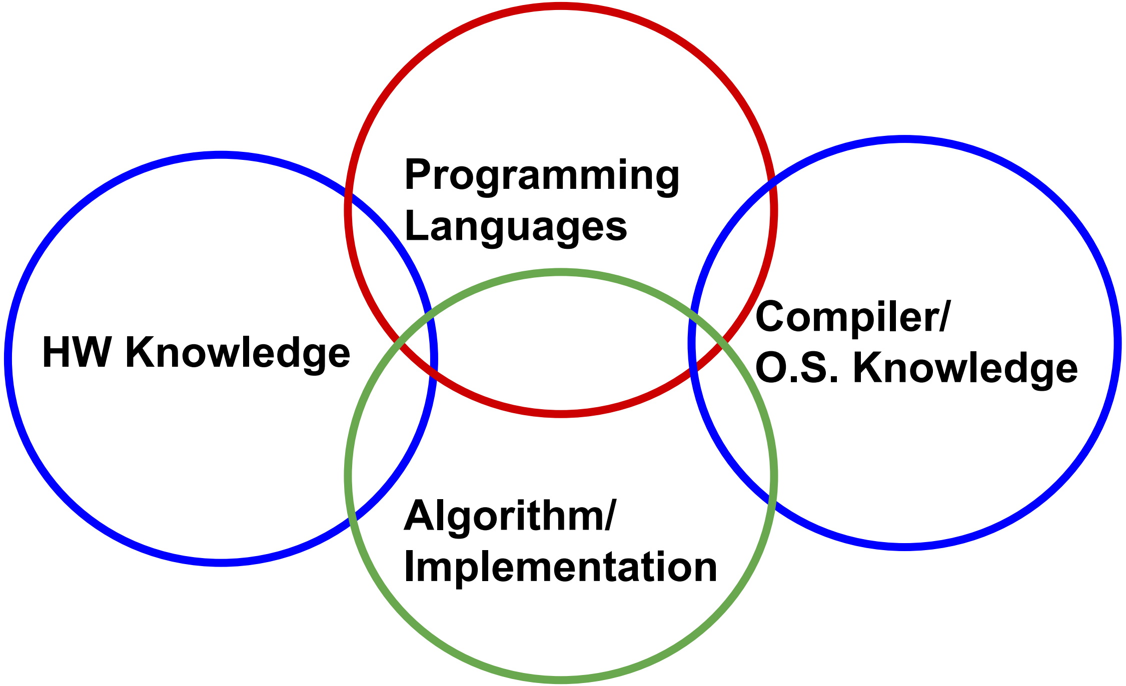
✓ High-performance computing (HPC) uses supercomputers and computer clusters to solve advanced computation problems

Personal definition:

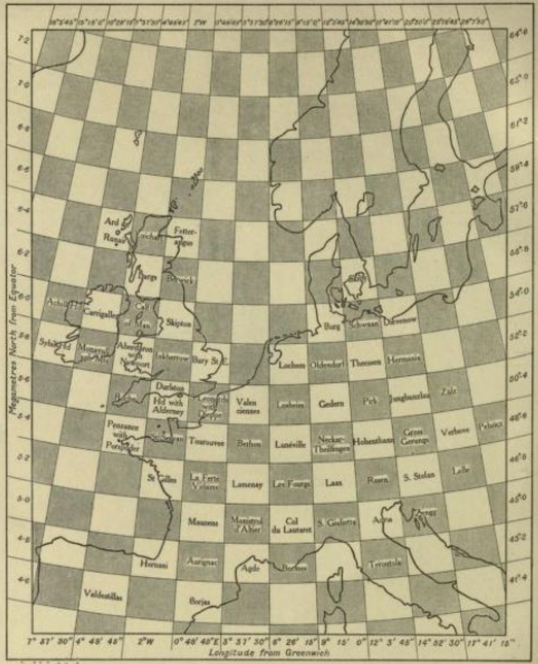
✓ It is the overlap of different skills, all devoted to exploit as much as possible the HW performance (both serial and/or parallel, but not limited to supercomputers…)

HPC: what it is?

✓ These are the main skills for an efficient HPC



HPC: older than computers?



An arrangement of meteorological stations designed to fit with the chief mechanical properties of the atmosphere. Other considerations have been here disregarded. Pressure to be observed at the centre of each shaded chequer, velocity at the centre of each white chequer. The numerical coordinates refer to these centres as also do the names, although as to the latter there may be errors of 5 or 10 km. The word "with" in "St Leonards with Dieppe" etc. is intended to suggest an interpolation between observations made at the two places. See page 9, and Chapters 3 and 7. Contrast the existing arrangement shown on p. 184.

WEATHER PREDICTION

BY

NUMERICAL PROCESS

BY

LEWIS F. RICHARDSON, B.A., F.R.MET.SOC., F.INST.P.

FORMERLY SUPERINTENDENT OF ESKDALEMUIR OBSERVATORY

LECTURER ON PHYSICS AT WESTMINSTER TRAINING COLLEGE

CAMBRIDGE

AT THE UNIVERSITY PRESS

1922

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 n^2 matrices we have to:

- ✓ compute n^3 products and n^3 sums
- ✓ load 2*n^2 data and to store n^3 data

```
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 multiplications are used for supercomputing rankings

Example: matrix-matrix multiplication/2

- ✓ Performance can really different, depending on HW, implementation etc….
- ✓ Improvement in performance can be really high….
- ✓ ….or you can easily “depress” performance
- ✓ Performance in Mflops: higher is 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**

Example: matrix-matrix multiplication/3

Simple problem

✓ Performance can really different, depending on HW, implementation etc….

#test	size	HW	Note	MFlops	Ratio
1	2048	CPU AMD EPYC 7742	cache unfriendly	201	-
2	2048	CPU AMD EPYC 7742	cache friendly	4870	24x
3	8192	GPU A100	do_concurrent	361328	1797x
4	8192	GPU A100	matmul	448923	2233x

Few “facts” about HPC

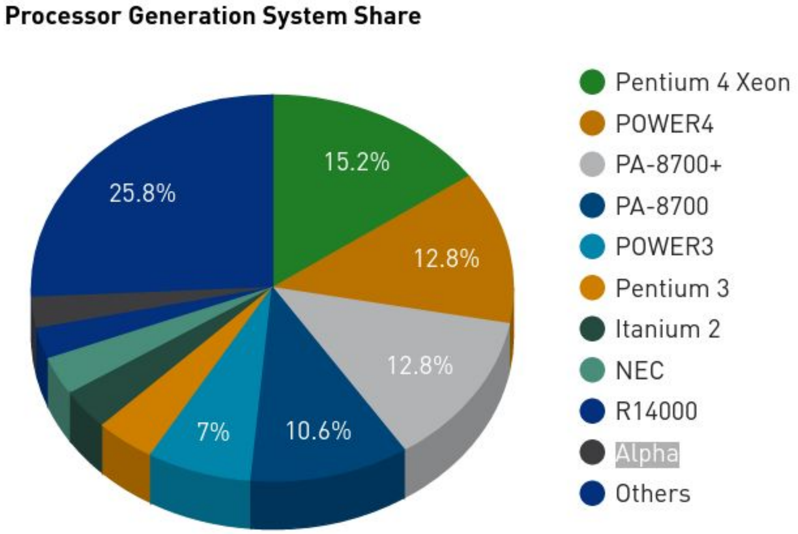
HPC market is not big enough to survive…

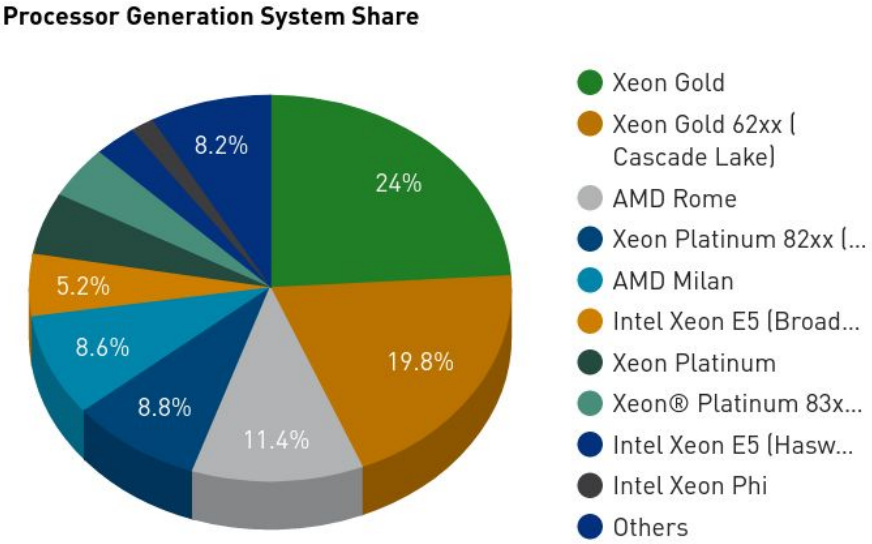
- ✓ SGI
- ✓ Compaq
- ✓ Digital
- ✓ SUN
- ✓ SiCortex
- ✓ MTA
- ✓ CRAY
- ✓ CONVEX
- ✓ CDC
- ✓ Thinking Machine
- ✓ Quadrics/APE

- ✓ IBM
  - Power3/4/.../9
- ✓ Intel
  - Itanium
  - Phi
- ✓ HP
- ✓ NVIDIA
- ✓ FUJITSU
- ✓ AMD
  - Opteron
- ✓ NEC
  - SX6

Few “facts” about HPC

Top500 list: June 2003 vs November 2022





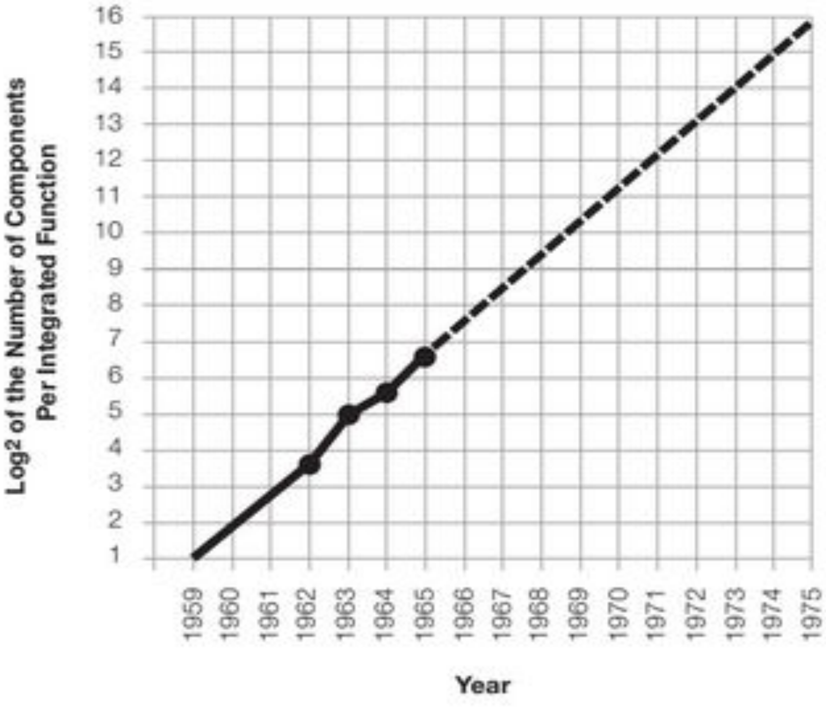
Moore's Law

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

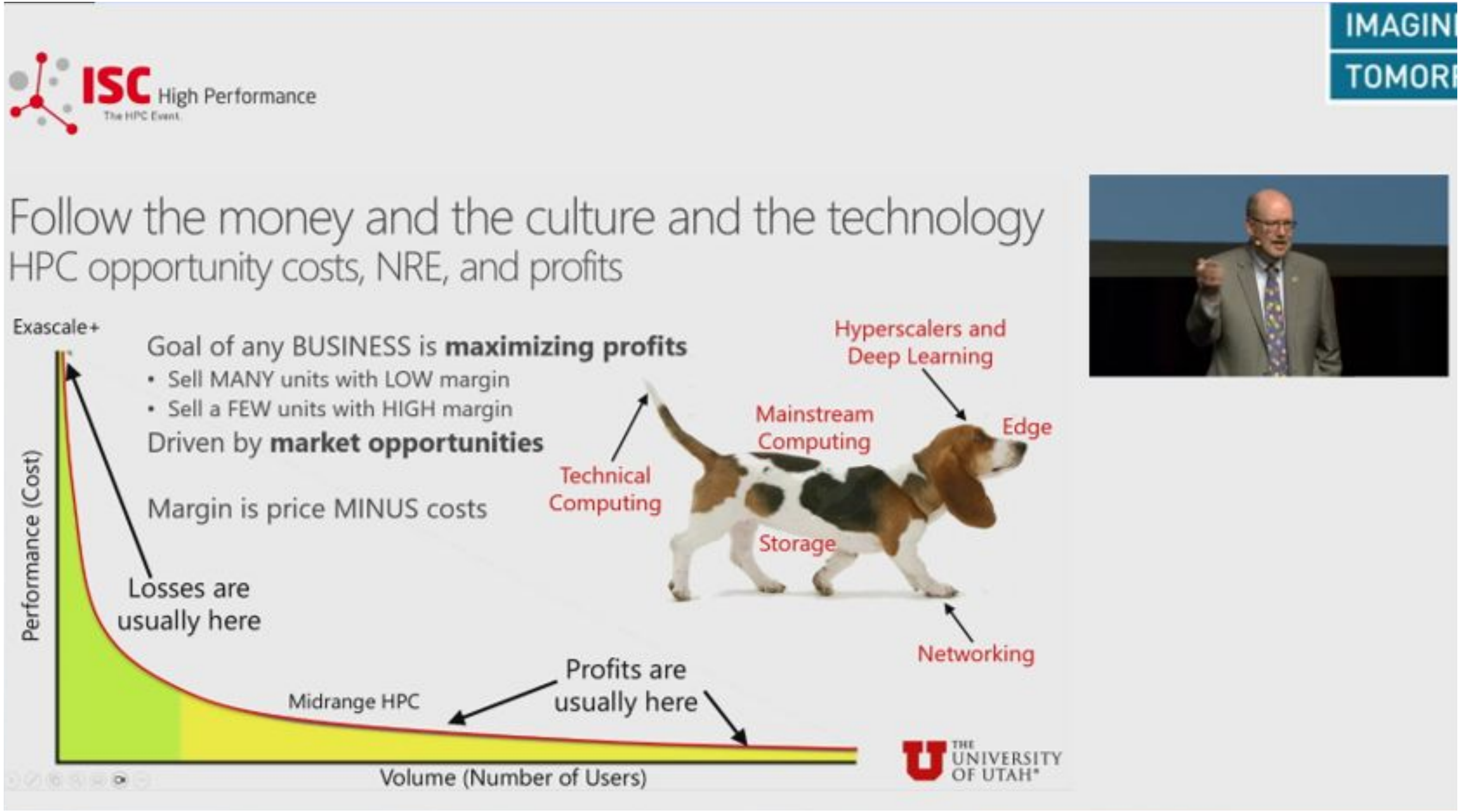
"With unit cost falling as the number of components per circuit rises, by 1975 economics may dictate squeezing as many as 65,000 components on a single silicon chip"

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

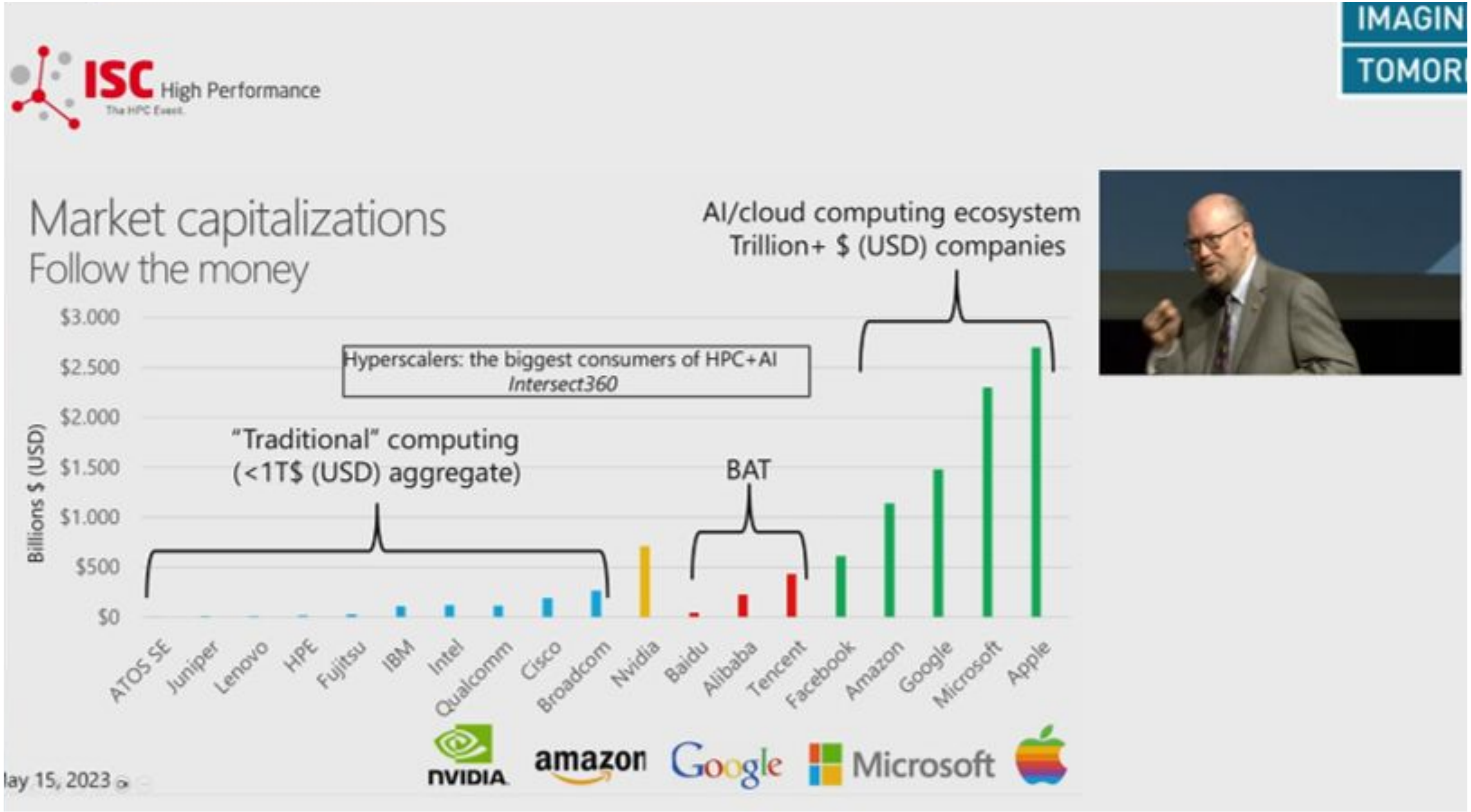
- ✓ This law is still valid now (in some form): to "survive" in the market HW firms must follow this law
- ✓ Now "transistor shrinking" is much harder: we are near to quantum effects
- ✓ New "ideas" must be found



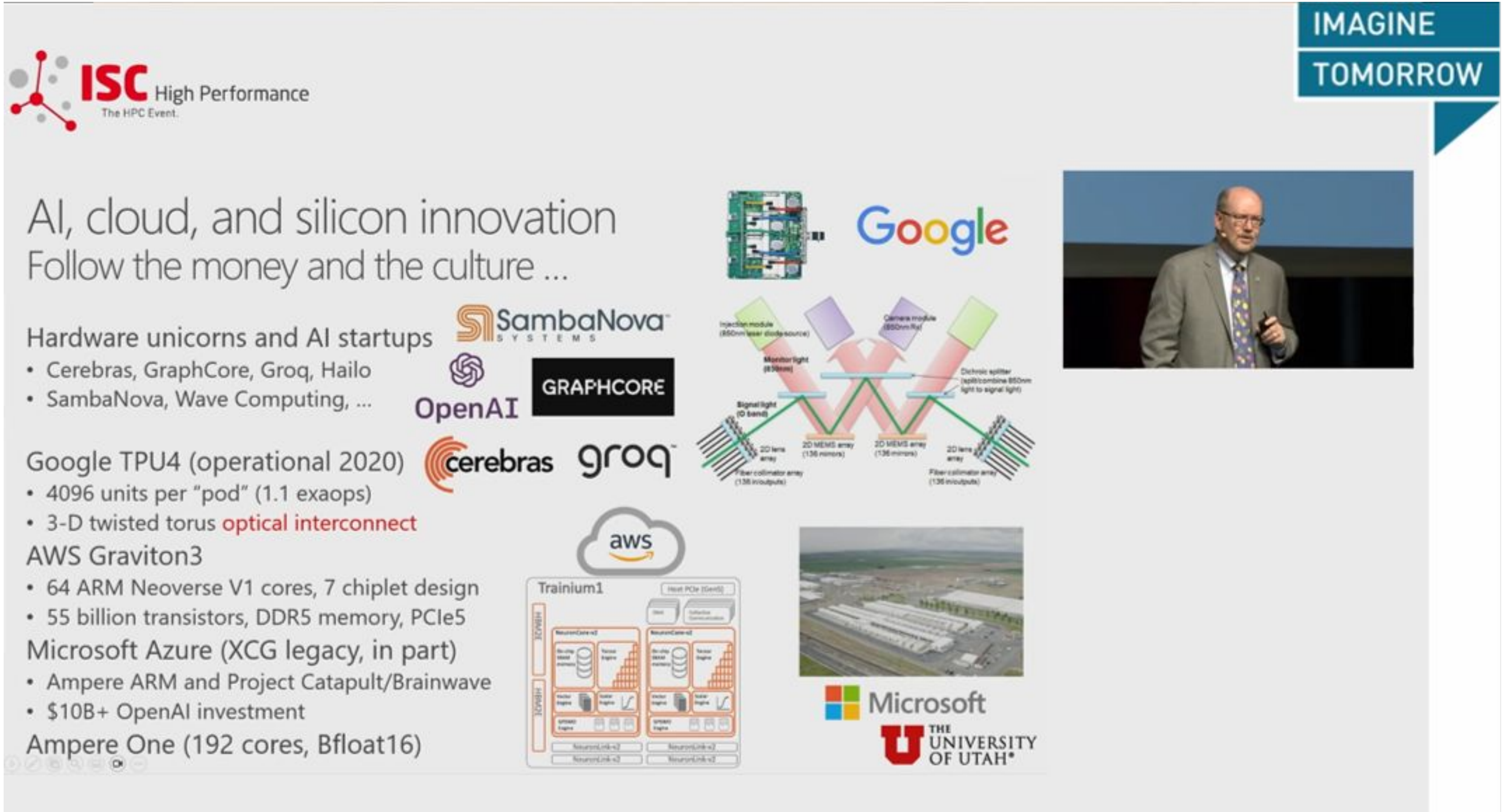
Follow the money! (from D.Reed@ISC2023)



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



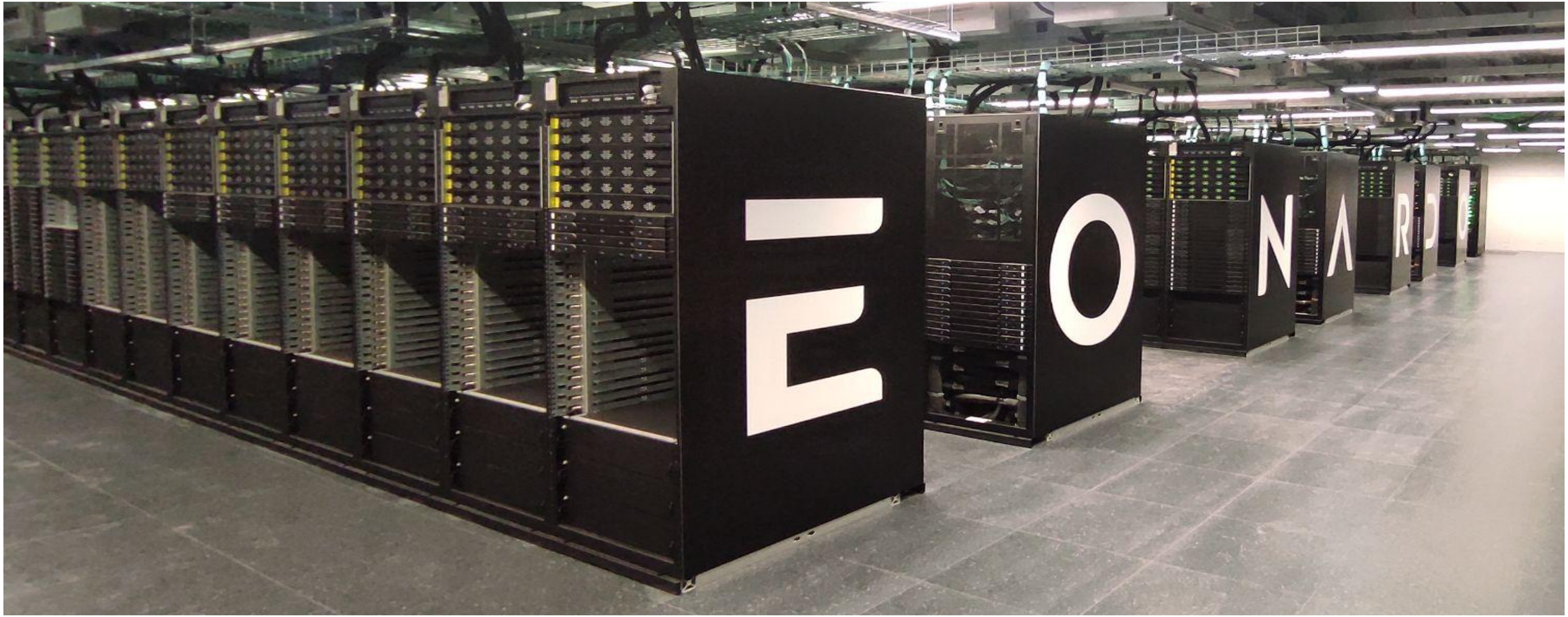
Follow the money! (from D.Reed@ISC2023)



Old stuff….

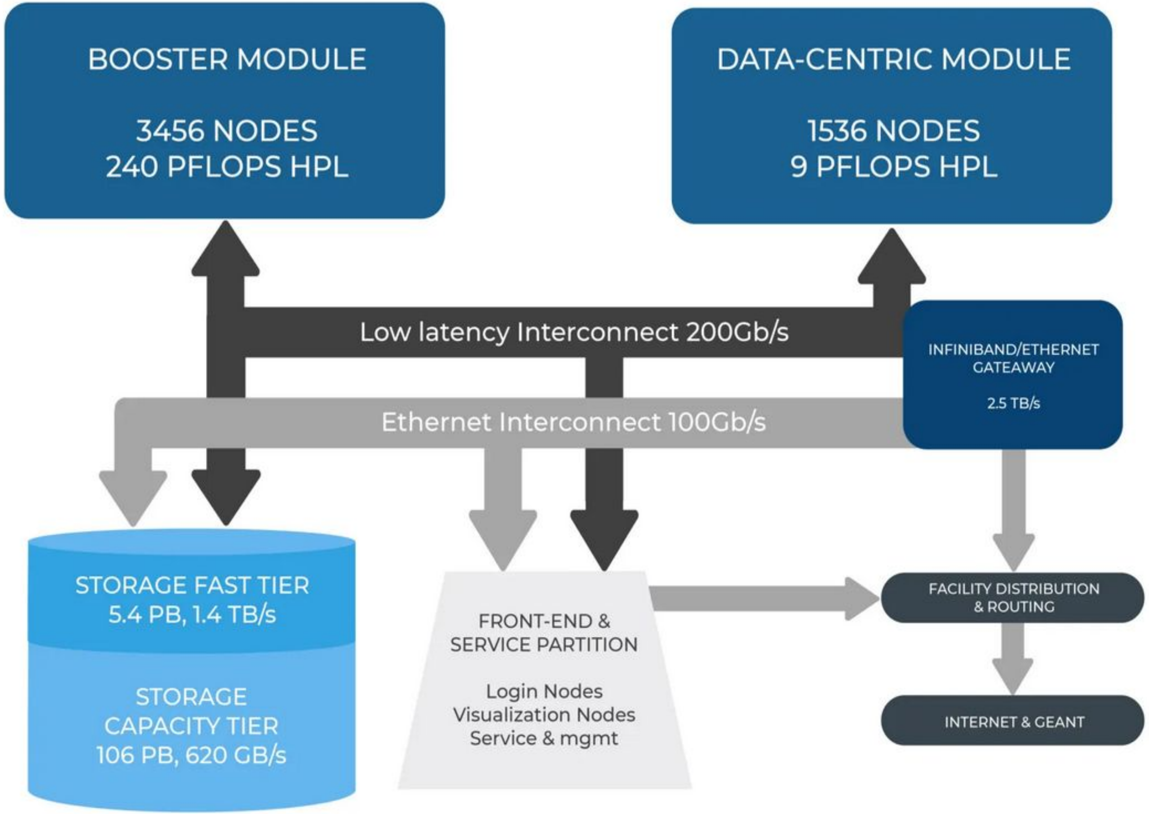

sgi
Altix 4700
2023/1/20 09:13
ORIGIN 2000
SiliconGraphics
ORIGIN 2000
2023/1/20 09:13
CONVEX COMPUTER CORPORATION
CONVEX
COMPAQ ALPHASERVER GS140
COMPAQ
2023/1/20 09:13

Leonardo

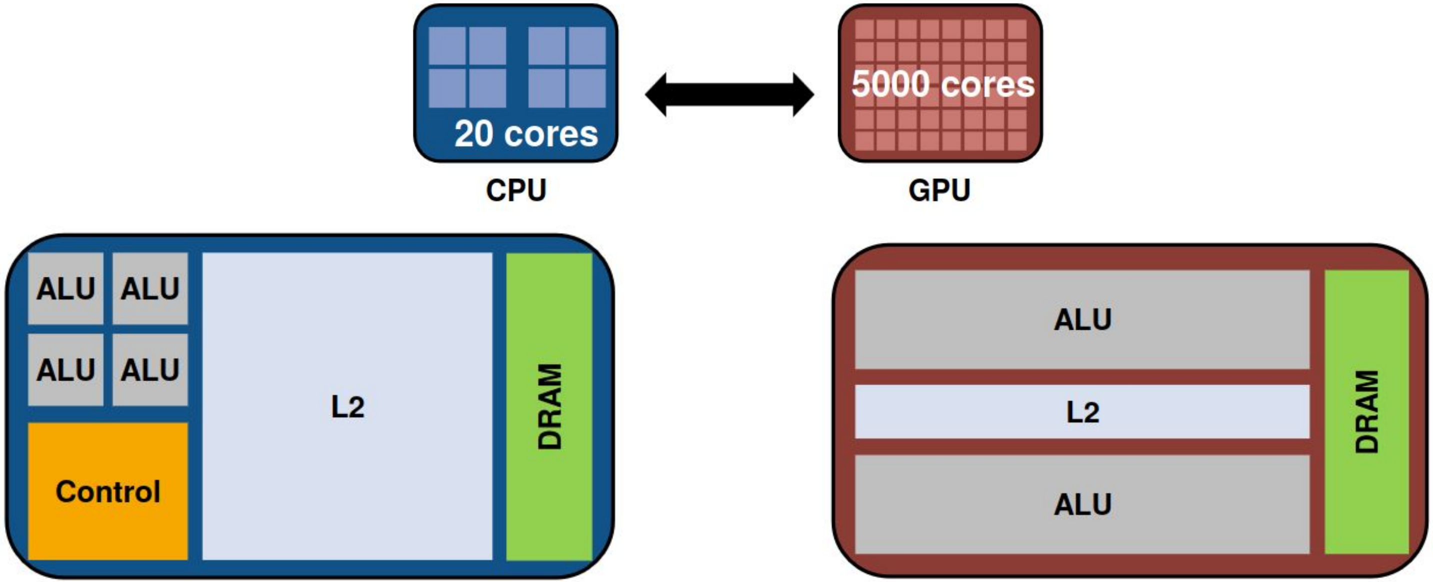


Leonardo: main figures

- ✓ 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**
  - GPU more powerful: 1 GPU ~ 10x CPU (Peak Mflops)
  - GPU ask for less space: for same performance CPU ask for ~3x more racks
  - GPU are less expensive: for same peak performance CPU are ~2x more expensive
  - GPU asks for (relative) less power: for the same peak performance CPU needs ~4x more energy
- ✓ **Cons**
  - GPU are less flexible respect CPU
  - 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

Recap

- ✓ Different skills are required to achieve “good” performance
- ✓ Performance is not only a problem of the right (powerful) HW
- ✓ HW evolution is driven by mass market
- ✓ All firms devoted only to HPC have not survived to the market
- ✓ User 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 soon optimization issues
  - 1 way to go fast, 100 ways to go slow!
- ✓ Today CPU/GPUs can have order of 100’000’000’000 transistors