

02614

High-Performance Computing

Bernd Dammann

Associate Professor

Scientific Computing DTU Compute

building 303B, room 104

<beda@dtu.dk>



Bernd Dammann

HPC Architect & Scientific Lead

HPC Competence Center
DTU Computing Center

building 324, room 280

<bd@cc.dtu.dk>

Course curriculum

Three modules:

- ❑ Serial tuning (week 1)
 - ❑ Parallel computing with OpenMP (week 2)
 - ❑ GPU computing with OpenMP (week 3)
-
- ❑ Three projects – one per week

Course Overview – Topics

- Hardware basics: CPU, caches, memory
- Tuning of sequential programs
- Compilers, Debuggers, Analysis Tools
- Libraries
- Parallel computers – multi-core, SMP, clusters
GPGPUs, etc
- Parallel Programming with OpenMP
- GPU computing with OpenMP

Practicalities – I

- ❑ Lectures, exercises, project work, etc:
 - ❑ “*Every day*”, 9 – 17
 - ❑ lectures, and labs/project work (check calendar on DTU Learn)
- ❑ Teachers (week 1 + 2):
 - ❑ Bernd Dammann <beda@dtu.dk>
 - ❑ TA: Rasmus
 - ❑ members of the HPC team (Sebastian, Andrea, Hans-Henrik)

Practicalities – I (cont'd)

- ❑ more participants (week 2):
 - ❑ students from course 41391 will join us for week 2
- ❑ more teachers (week 3):
 - ❑ Hans-Henrik Sørensen <hhbs@dtu.dk>
 - ❑ TA: Rasmus
 - ❑ members of the HPC team (Sebastian, Andrea, Bernd)

Practicalities – II

- ❑ Lecture notes:
 - ❑ will be made available on DTU Learn
- ❑ Exercises:
 - ❑ material on DTU Learn
 - ❑ access to DTU Linux computers via SSH or ThinLinc
- ❑ On-line updates:
 - ❑ last minute info will be published on DTU Learn

Practicalities – III

Literature:

- ❑ Part I – Serial Tuning:
 - ❑ “Introduction to High-Performance Scientific Computing” by Victor Eijkhout, U of Texas and TACC – on-line available (also as PDF): <https://theartofhpc.com/istc.html>
 - ❑ “Introduction to High Performance Computing for Scientists and Engineers”, by G. Hager & G. Wellein, CRC Press (on-line [via DTU Library](#)), Chapters 1-8 relevant for this course
 - ❑ other relevant references will be made available during the course

Practicalities – III (cont'd)

Literature:

❑ Part II – OpenMP:

- ❑ on-line references and articles
- ❑ “Using OpenMP – The Next Step“ by R. van der Pas, E. Stotzer and C. Terboven, MIT Press (2017) (via DTU Library)
- ❑ “The OpenMP Common Core” by T.G. Mattson, Y. He, and A.E. Koniges, MIT Press (2019) (via DTU Library)
- ❑ “Using OpenMP – portable shared memory parallel programming” by B. Chapman, G. Jost and R. van der Pas, MIT Press (2008)
- ❑ Hager & Wellein (see week 1)

Practicalities – III (cont'd)

Literature:

- ❑ Part III – GPU computing with OpenMP
 - ❑ more details in week 3

Practicalities – IV

- ❑ Three assignments:
 - ❑ Groupwork: 3-4 students/group
 - ❑ Note: 1 student is NOT a group!
 - ❑ Assignment I: Serial tuning
 - ❑ deadline: Friday, January 9, 16:00 (!!)
 - ❑ Assignment II: OpenMP
 - ❑ deadline: Friday, Jan 16, midnight
 - ❑ Assignment III: GPU computing
 - ❑ deadline: Friday, Jan 23, midnight

Practicalities – IVa

- ❑ “Individualized reports”:
 - ❑ each assignment has four sub-tasks
 - ❑ when handing in the reports, each group member has to take main responsibility for one sub-task
 - ❑ this should be documented in an addendum to the report (we provide a template)
- ❑ But ... remember: all group members should contribute equally to the work, and should know about all parts of the assignment!

Practicalities – V

Requirements for this course:

- ❑ Knowledge of at least one of these programming languages: C/C++ (or Fortran)
- ❑ Note: Python or MATLAB are not enough!
- ❑ Basic understanding of numerical computations
- ❑ The will to “play” with new tools and to explore new fields on your own.
- ❑ To be able to document what you have done.

Practicalities – VI

Computer usage:

- ❑ You are encouraged to use the DTU computer systems – at least for your “production runs”
- ❑ Especially needed in weeks 2 and 3, but make your first steps in week 1, already!
- ❑ Well defined environment – that is known to work
- ❑ Same environment for everybody
- ❑ Don't waste time to “roll your own”

Practicalities – VII

Lab exercises & projects:

- ❑ Please do the labs! They are the foundations for the projects/assignments
- ❑ Read the assignments carefully – and follow the instructions
- ❑ We do experiments, and our lab is the computer
- ❑ Describe your findings in a well written report – see the 'Assignment Guide' on DTU Learn

What is High-Performance Computing?

What is HPC?

Do you want to be in low performance computing?



How do I get from A to B as fast as possible?

Vehicle A:



Vehicle B:



Vehicle C:



Your choice:

A, B or C?

Road X:



Road Y:



Road Z:



Your choice:

What now?

Payload 1:



Payload 2:



Payload 3:



Payload 4:



Your choice:

Help – there are (too) many choices ...

... and what has this to do with
High-Performance Computing?

How do I get from A to B as fast as possible?

... or:

How do I get from my problem (A) to a solution (B) as fast as possible?

(Large Scale) Computations

- ❑ Computers



- ❑ Algorithms/
Codes



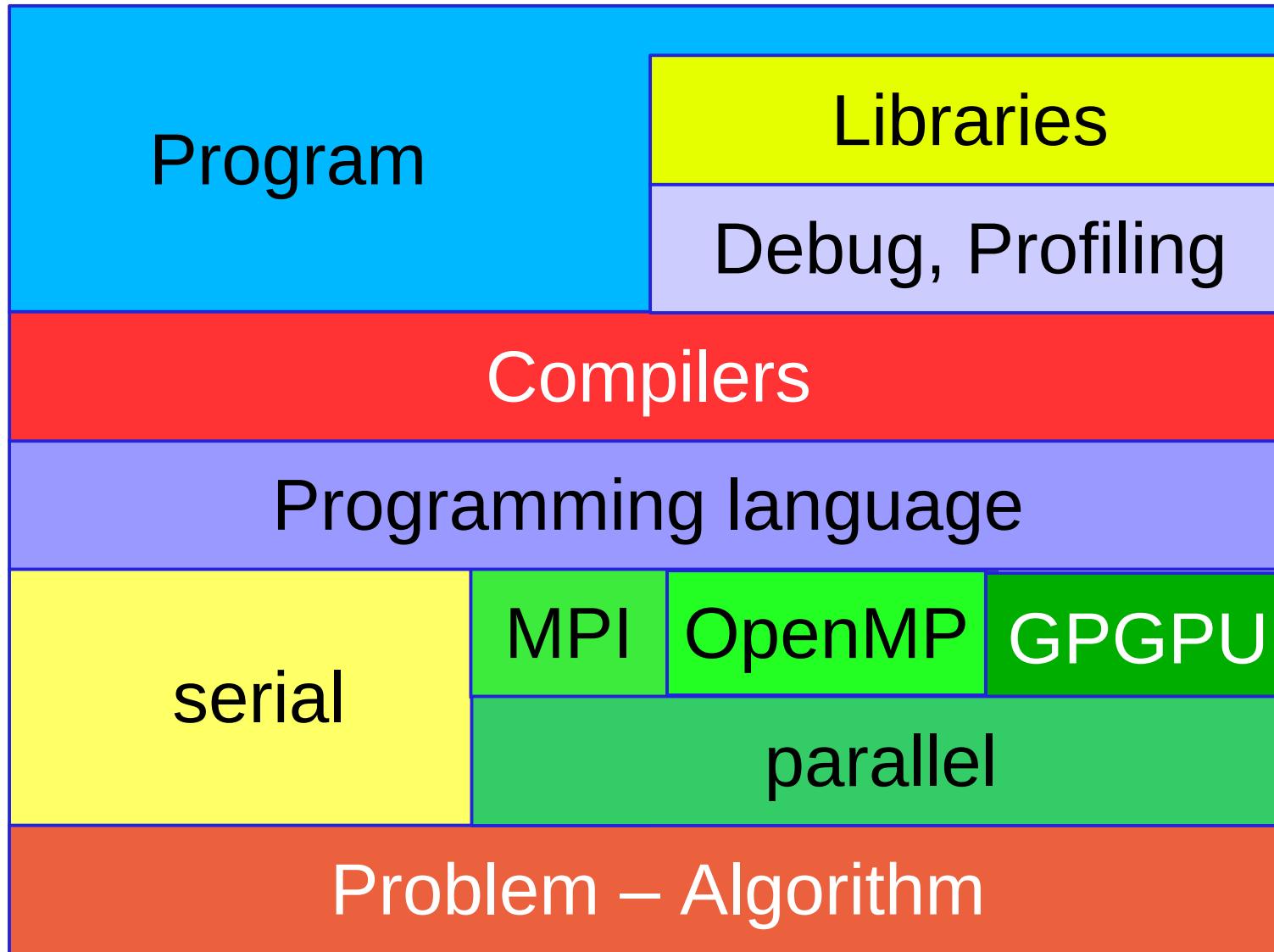
- ❑ Data



Large Scale Computations



HPC's Caterpillar



... and not to forget:



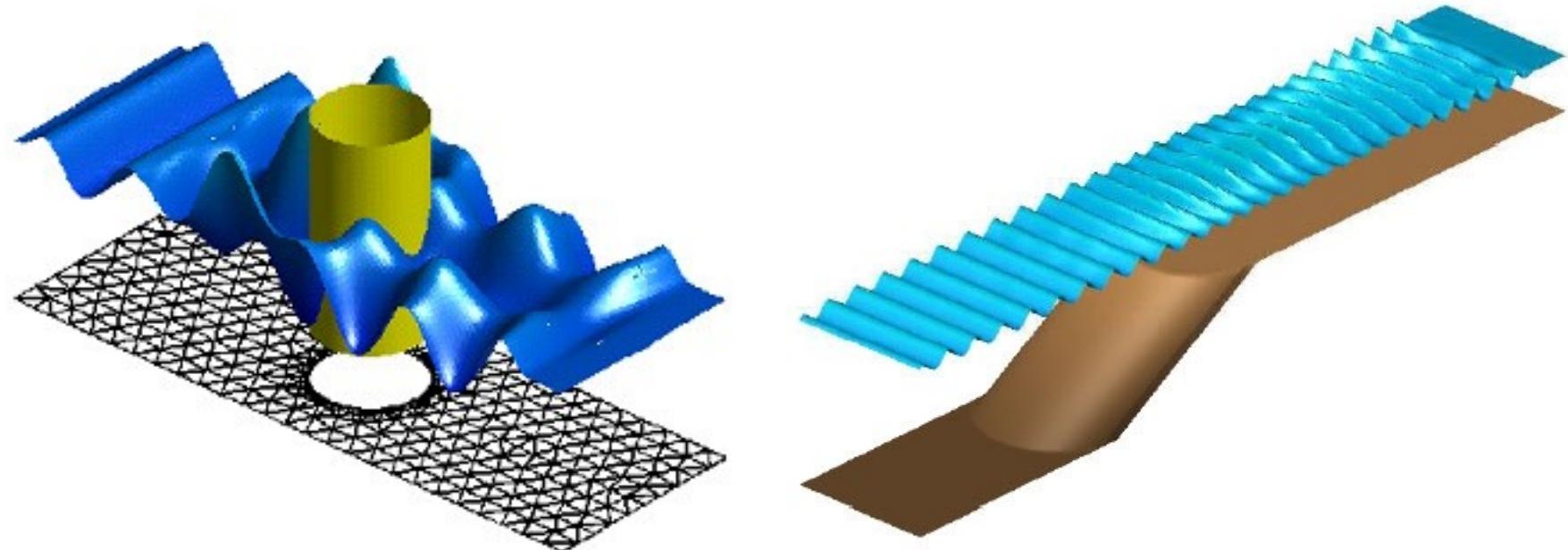
Computer Simulations

- ❑ Alternative to scale models and lab experiments
 - ❑ faster and cheaper – more flexible
- ❑ Allows a variety of studies
 - ❑ isolated phenomena
 - ❑ change of one parameter at a time
- ❑ Realistic models are large
 - ❑ many model parameters
 - ❑ capture fine details – fine discretization
 - ❑ simulation over a long period of time

Scientific Computing – Examples

- **Astrophysics**
 - stellar physics
 - galaxy evolution
 - exoplanets
- **Cryptography**
 - prime numbers
- **Data mining**
 - Google's Page rank
 - BIG DATA
- **Planetary science**
 - geophysics
 - weather forecasts
 - air pollution
 - climate modeling
- **Quantum Physics & Chemistry**
 - superconductivity
 - material science
 - enzymes
- **Bio-informatics**
 - genome research
 - neuroscience
 - heart simulation
- **Engineering design**
 - fluid mechanics, turbulence
 - hydro dynamics
 - structural design
- **Finance (FinTech)**
- **Machine Learning**
- **AI**

Breaking The Waves



Allan Engsig-Karup – DTU Compute

Wind turbine design - CFD



DTU Wind Energy

Topology Optimization

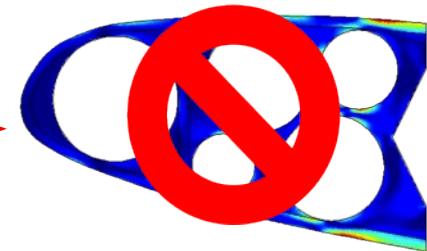
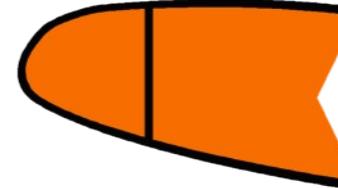
... and Materials:

safe and minimum weight structures



DTU Mechanical
Engineering

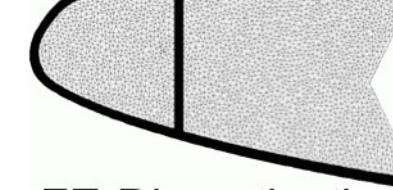
Topology Optimization



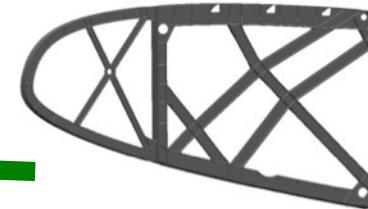
Design domain



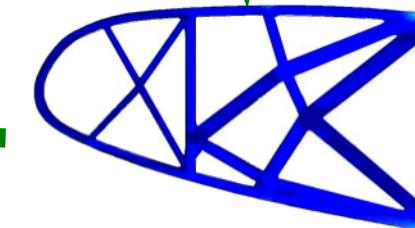
courtesy: Ole Sigmund
www.topopt.dtu.dk



FE-Discretization



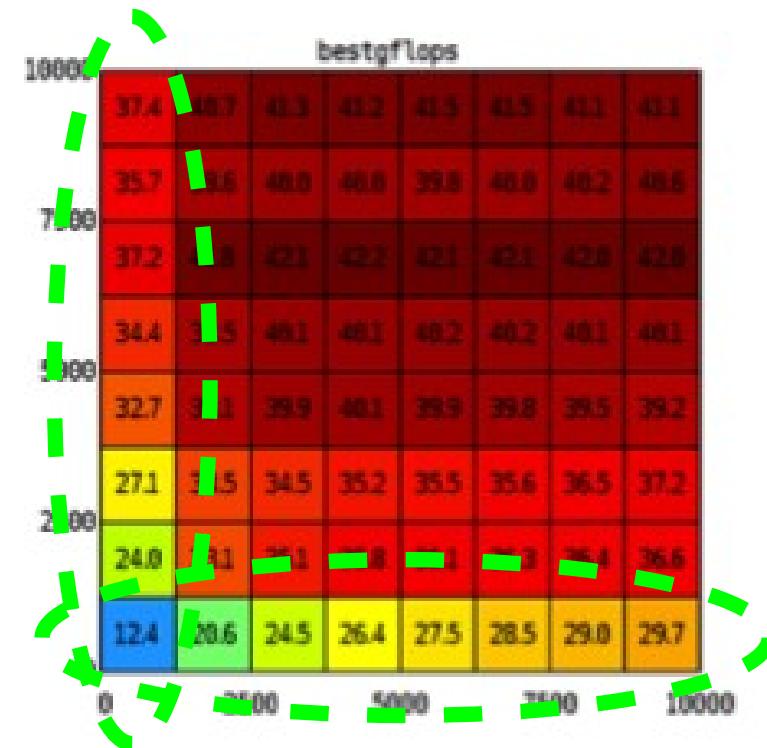
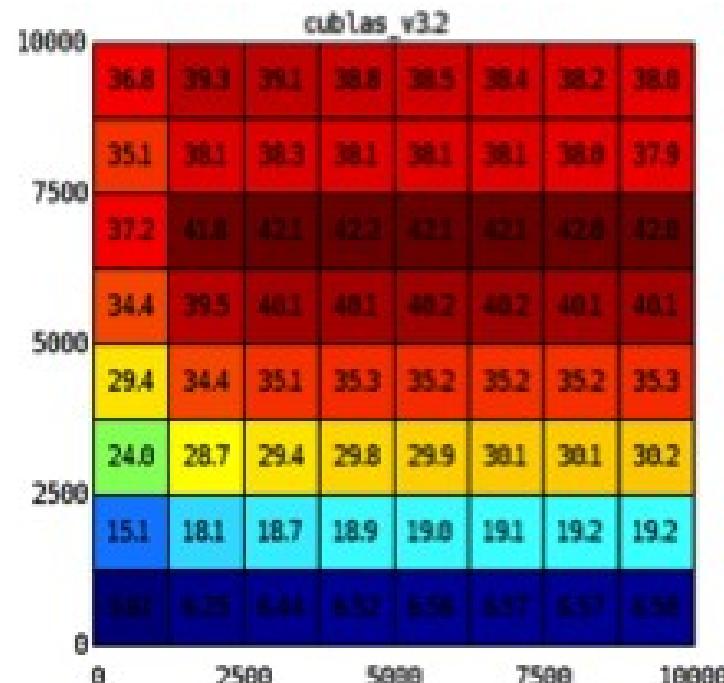
Interpretation



Simulation
result

Performance Tuning – GPU

Auto-tuning $Ax=y$ (Sgemv) on Nvidia Tesla C2050 (blue = slow, red = fast)

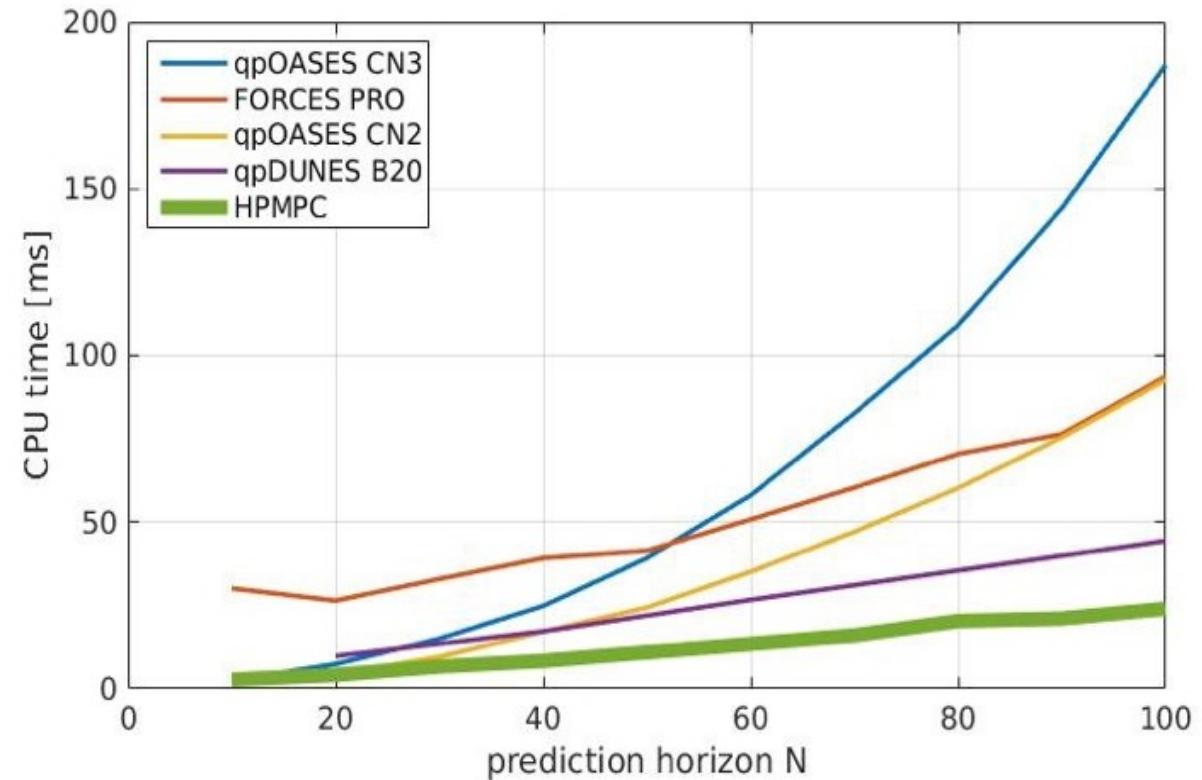


Hans-Henrik Sørensen – GPULab, DTU Compute

Model Predictive Control (MPC)

HPMPC:

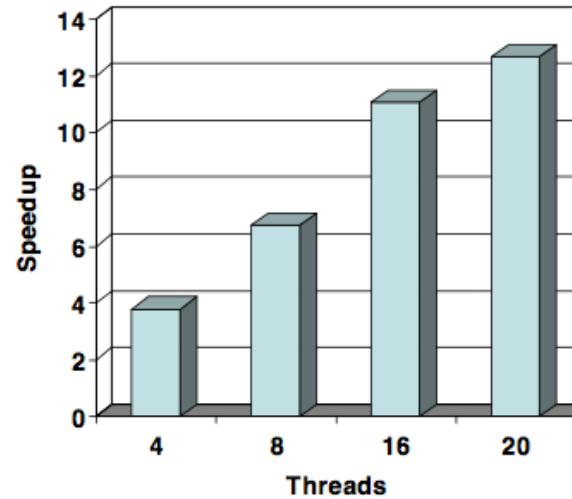
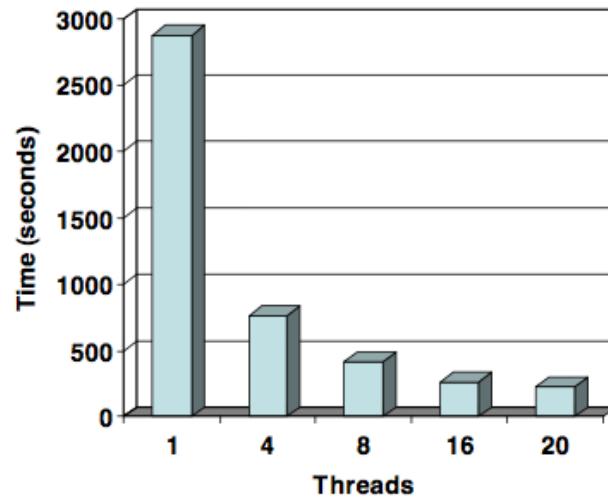
- ❑ optimized for small datasets
- ❑ “applied HPC”
- ❑ close to CPU peak performance



Gianluca Frison, et al – Scientific Computing, DTU Compute

Tuning & Parallelization

- ❑ Tuning and parallelization of an existing code from DTU Chemistry: Helium Scattering
- ❑ ~3000 lines of Fortran77 code
- ❑ parallelized with OpenMP



What is Performance?

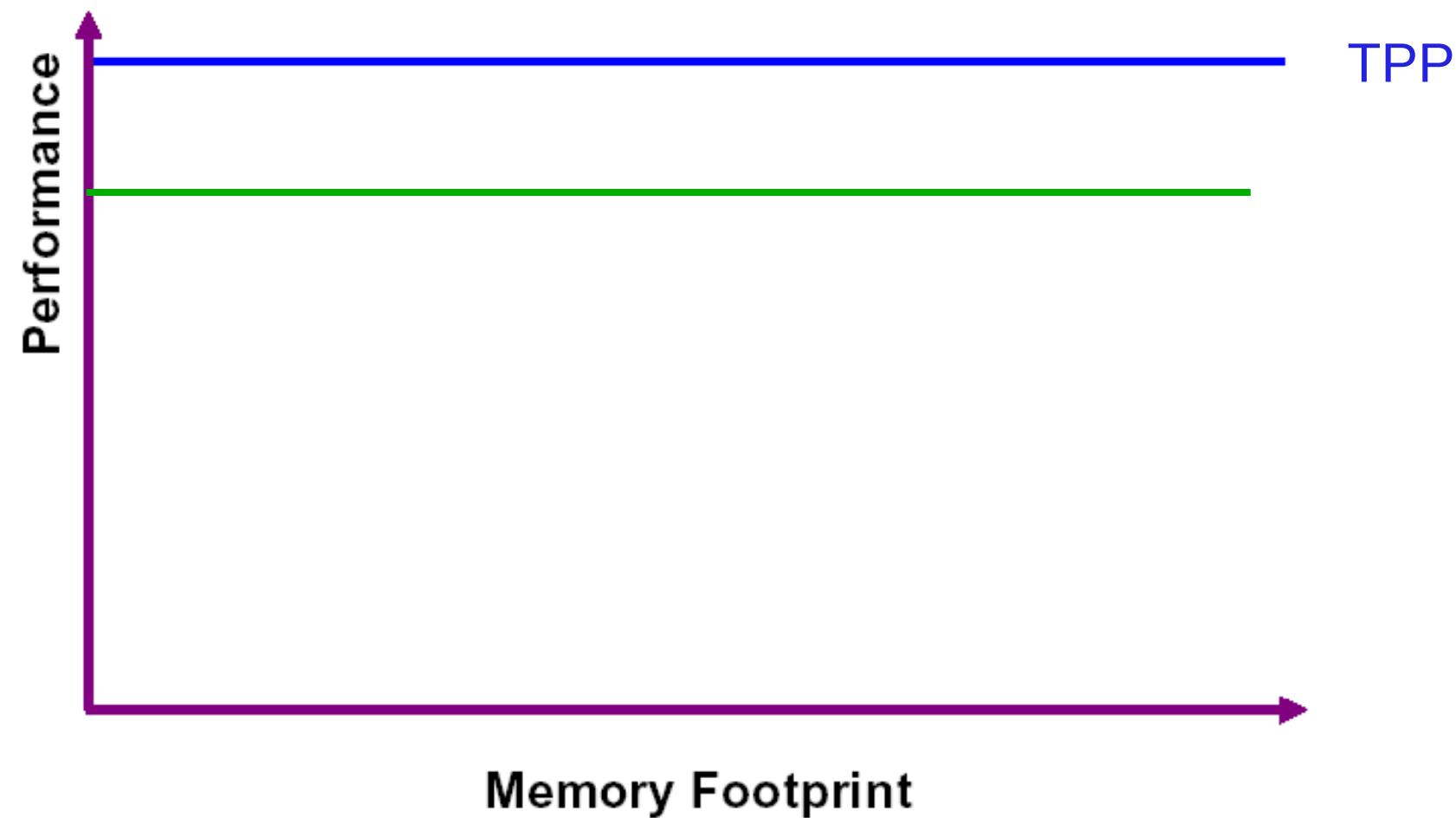
Performance of a Computer

- ❑ The performance of a computer is often expressed in Flop/s (floating point operations per second)
- ❑ How does this relate to the clock frequency of the CPU?
- ❑ Example:
 - ❑ modern Intel/AMD CPU, running at 2GHz
 - ❑ 16 Floating Point Ops per tick (double precision)
 - ❑ Performance: 32 GFlop/s

Theoretical Peak
Performance!!!

Performance of a Computer

Intuitive Performance Graph for a given problem:



Live test

- ❑ Measure timings/performance of two operations, matrix multiplication and matrix times vector – for different matrix sizes
- ❑ Use either ...
 - ❑ MATLAB
 - ❑ Python
- ❑ on your own computer
- ❑ ... or on the DTU systems (via ThinLinc)
- ❑ download code from DTU Inside

Live test

- ❑ MATLAB code

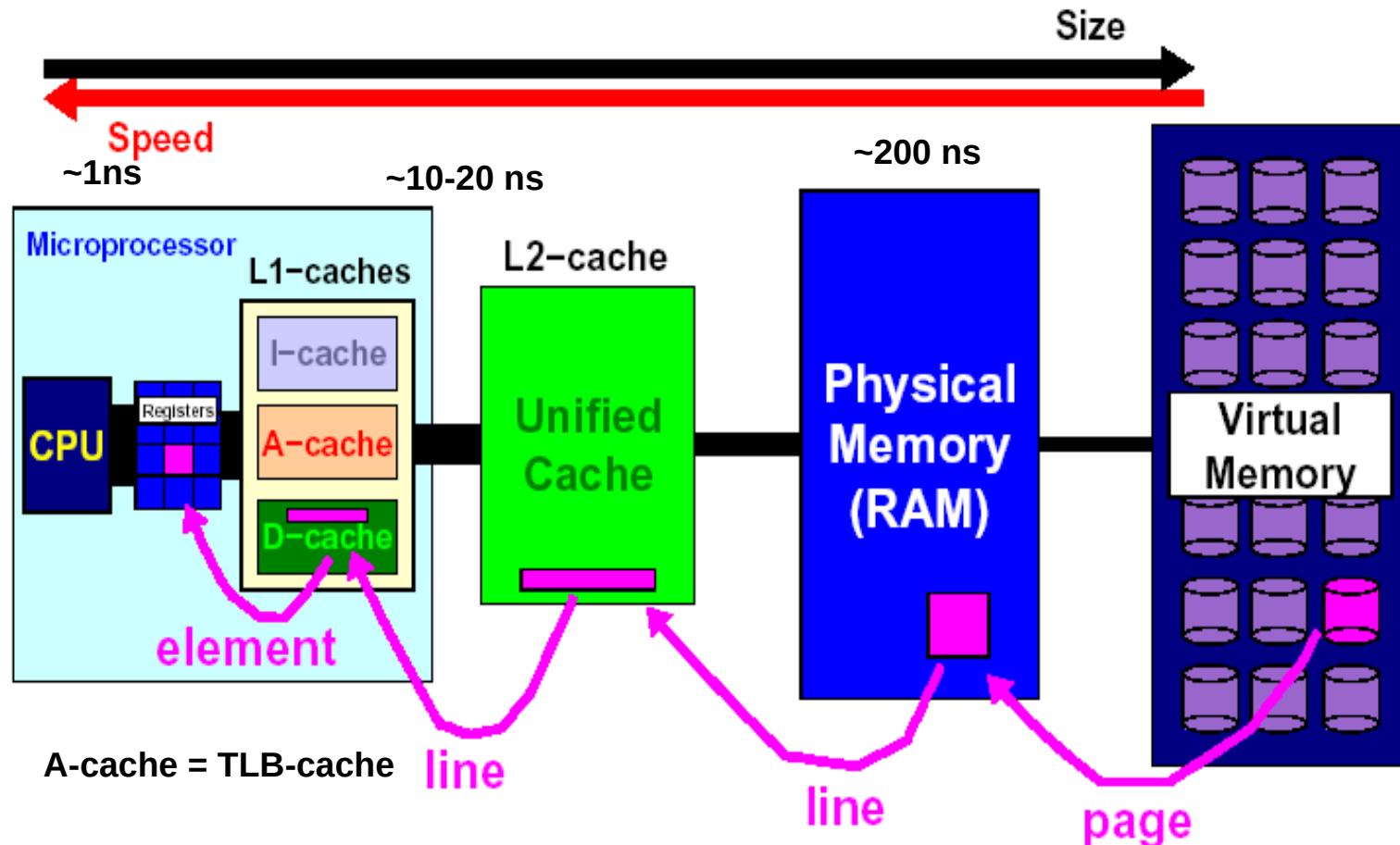
```
A = ones(n);  
tic;  
B = A*A;  
toc
```

- ❑ Python code

```
A = np.ones((n, n))  
t0 = time()  
B = A @ A  
dt = time() - t0
```

- ❑ do the above for different values of n (500..5000), and plot timings and performance (Mflop/s)
- ❑ use the example code from DTU Inside
- ❑ give it a try – and report results

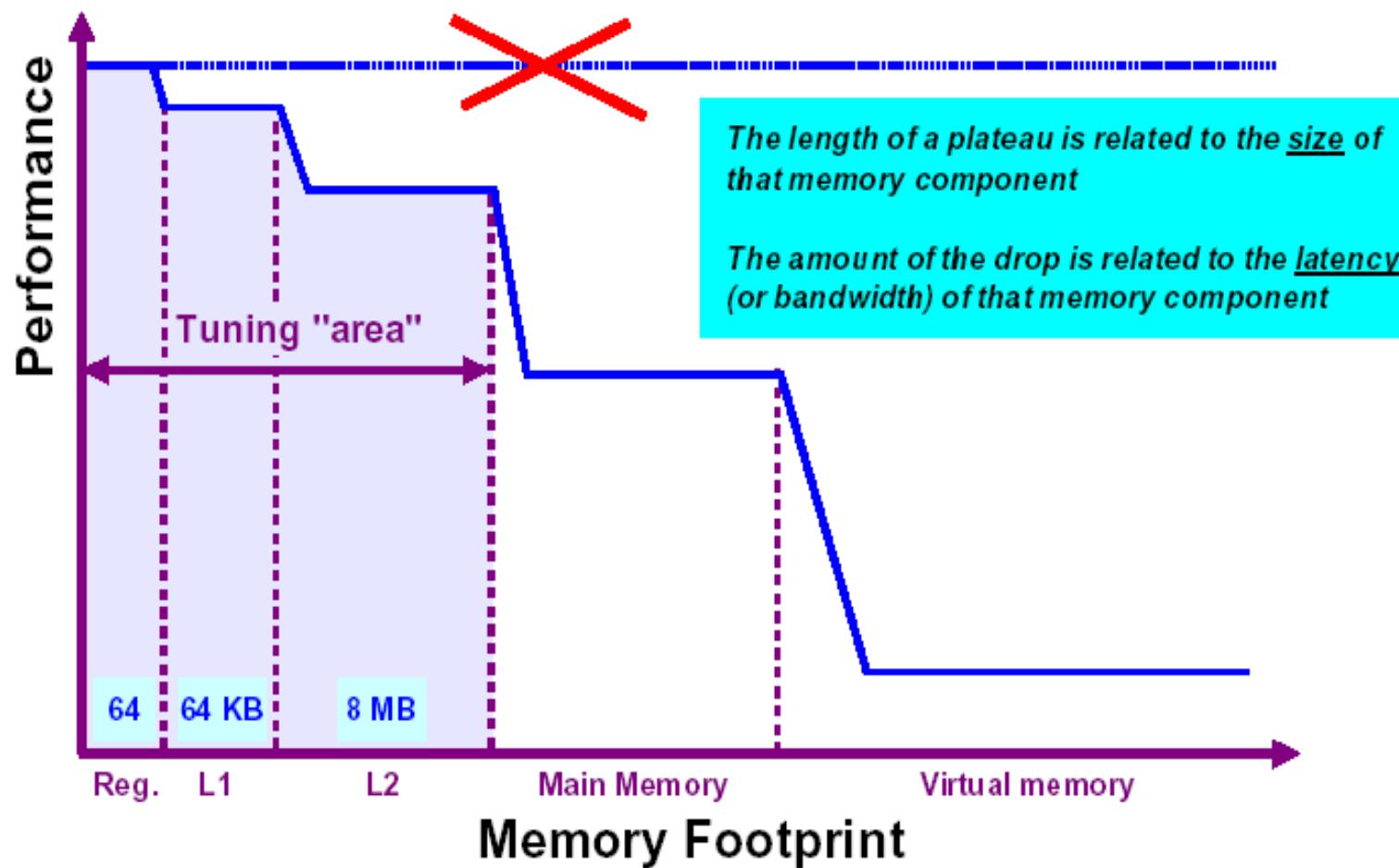
The Memory Hierarchy



*Memory Optimization:
Keep frequently used data close to the processor*

Performance of a Computer

Performance is not uniform:



TOP500 – HPC's Formula 1

- The “fastest” computers of the world are ranked on the TOP500 list

<http://www.top500.org/>



- Ranking is based on the High-Performance LINPACK (HPL) benchmark, i.e. a collection of linear algebra routines.
- Most of the top sites make use of special hardware, e.g. GPUs, i.e. hardware that is optimized to work with (dense) matrix data.

TOP 500 No. 1 – Nov 2010

Tianhe-1A (China)

- ❑ 7168 nodes
- ❑ 14336 Intel X5670
- ❑ 7168 Nvidia M2050
- ❑ 2048 FT-1000
- ❑ 112 racks
- ❑ ~ 4MW power
- ❑ 2.5 Pflop/s
- ❑ 4.7 Pflop/s (TPP)



TOP 500 No. 1 – Nov 2011

K Computer (Japan)

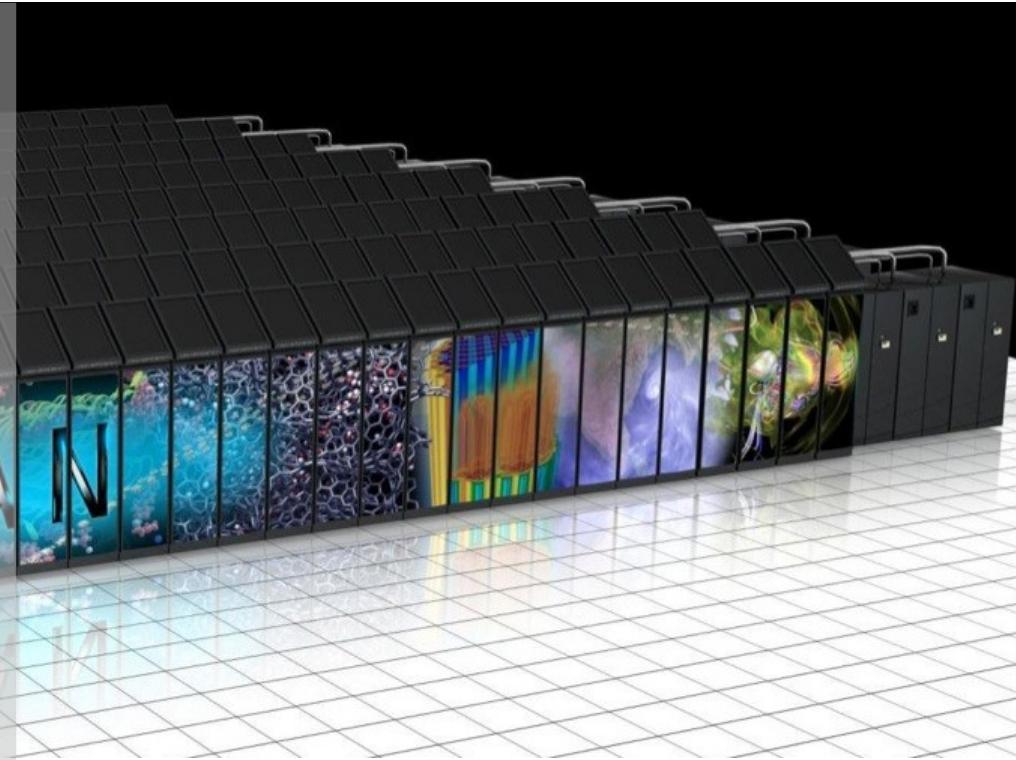
- ❑ 88128 SPARC64 VIIIfx
- ❑ 864 racks
- ❑ ~ 12 MW power
- ❑ 10.5 Pflop/s
- ❑ 11.3 Pflop/s (TPP)
- ❑ already #1 in June '11
- ❑ ~8 Pflop/s
- ❑ expandable



TOP 500 No. 1 – Nov 2012

Titan – Oak Ridge
(USA)

- ❑ Cray XK7
- ❑ Opteron + Nvidia K20
- ❑ 560640 cores
- ❑ 17.59 Pflop/s
- ❑ 27.11 Pflop/s (TPP)
- ❑ ~ 8.2 MW power



TOP 500 No. 1 – Jun 2016

Sunway TaihuLight
(China)

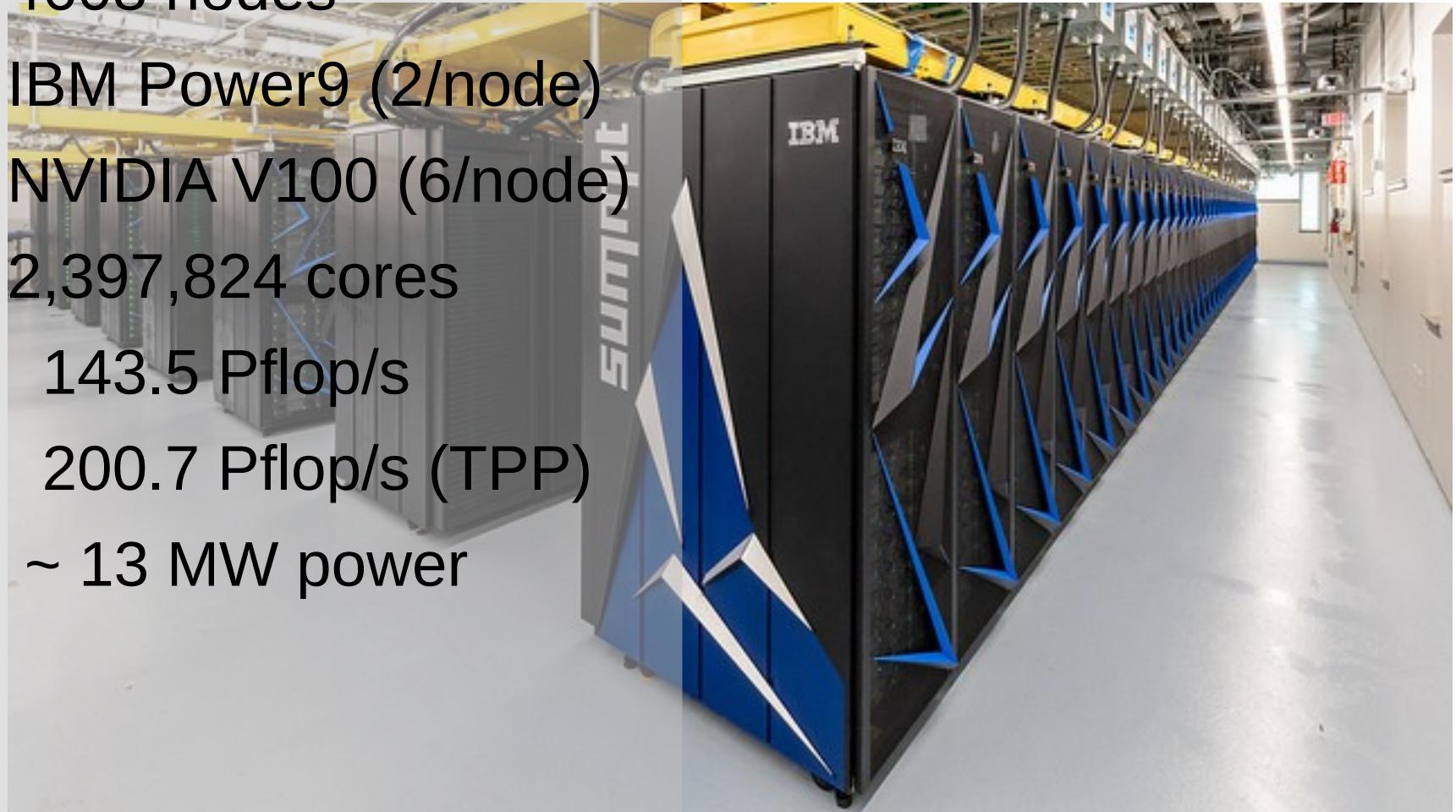
- Sunway SW26010 260C 1.45GHz
- 10,649,600 cores
- 93 Pflop/s
- 125 Pflop/s (TPP)
- ~ 15.4 MW power



TOP 500 No. 1 – Nov 2018

Summit (USA)

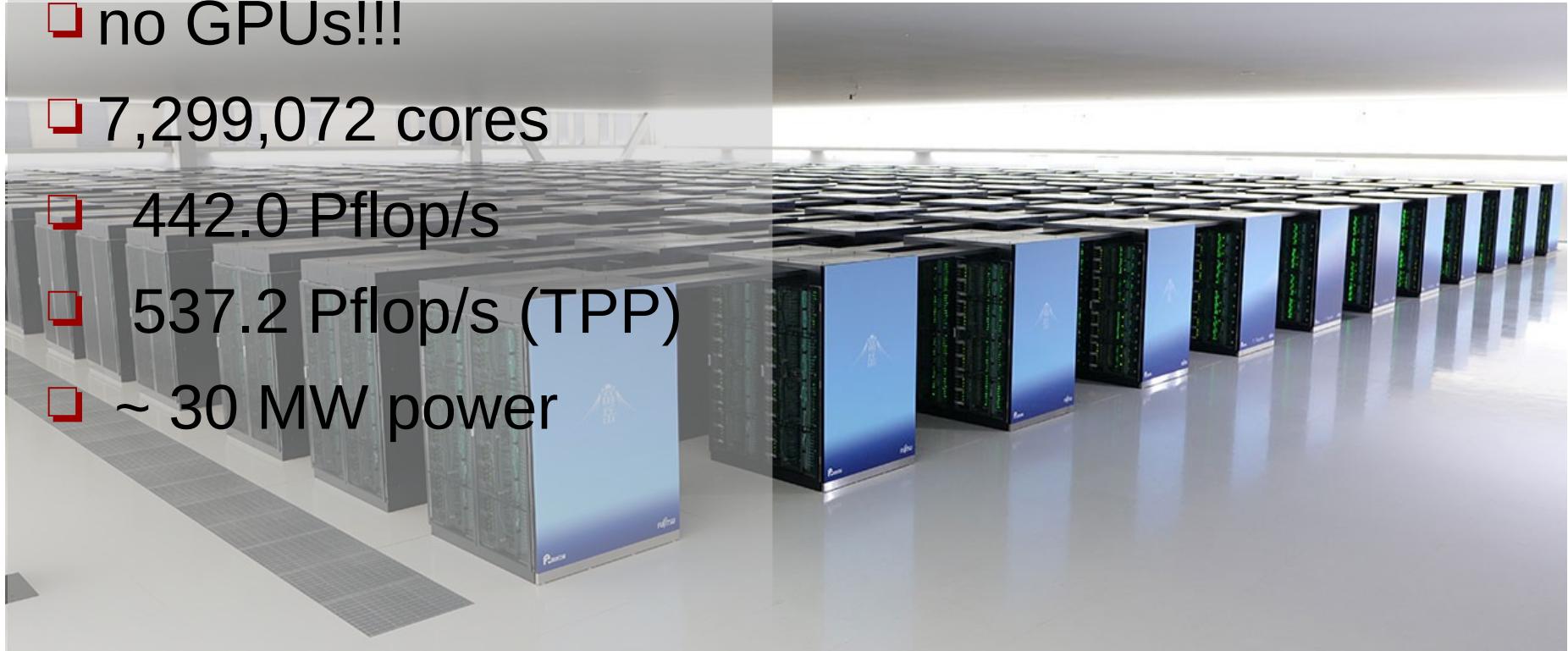
- ❑ 4608 nodes
- ❑ IBM Power9 (2/node)
- ❑ NVIDIA V100 (6/node)
- ❑ 2,397,824 cores
- ❑ 143.5 Pflop/s
- ❑ 200.7 Pflop/s (TPP)
- ❑ ~ 13 MW power



TOP 500 No. 1 – Jun 2020

Fugaku (Japan)

- ❑ 158,976 nodes
- ❑ ARM A64FX (48 cores)
- ❑ no GPUs!!!
- ❑ 7,299,072 cores
- ❑ 442.0 Pflop/s
- ❑ 537.2 Pflop/s (TPP)
- ❑ ~ 30 MW power



TOP 500 No. 1 – Jun 2022

Frontier (USA)

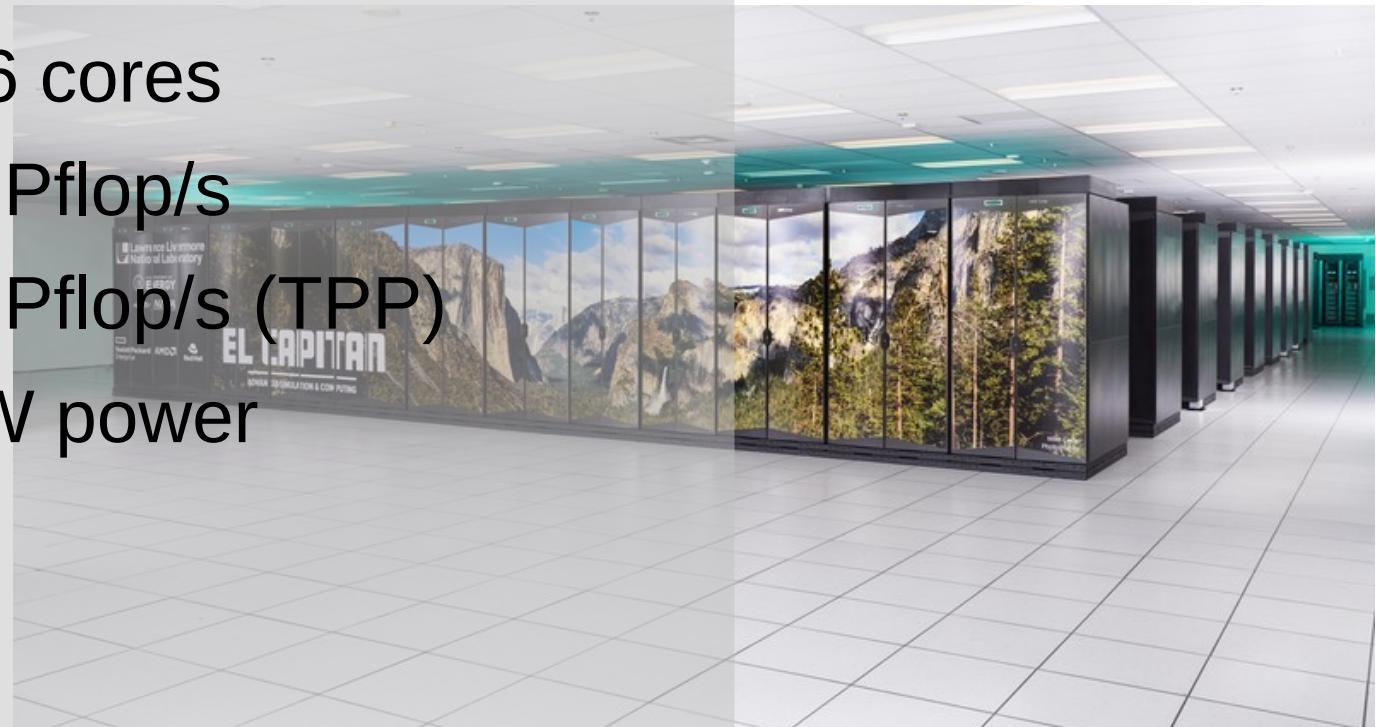
- ❑ 9,248 nodes
- ❑ 1 AMD EPYC (64 cores)
- ❑ 4 AMD Instinct MI250x GPUs
- ❑ 8,730,112 cores
- ❑ 1,102.00 Pflop/s
- ❑ 1,685.65 Pflop/s (TPP)
- ❑ ~ 21 MW power



TOP 500 No. 1 – Nov 2024

El Capitan (USA)

- ❑ 10,952 nodes
- ❑ 4 AMD MI300A APUs
 - ❑ 24 CPU cores + 6 GPUs
- ❑ 11,039,616 cores
- ❑ 1,742.00 Pflop/s
- ❑ 2,746.38 Pflop/s (TPP)
- ❑ ~ 29.6 MW power



TOP 500 – November 2025

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	El Capitan - HPE Cray EX255a, AMD 4th Gen EPYC 24C 1.8GHz, AMD Instinct MI300A, Slingshot-11, TOSS, HPE DOE/NNSA/LLNL United States	11,340,000	1,809.00	2,821.10	29,685
2	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE Cray OS, HPE DOE/SC/Oak Ridge National Laboratory United States	9,066,176	1,353.00	2,055.72	24,607
3	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	9,264,128	1,012.00	1,980.01	38,698
4	JUPITER Booster - BullSequana XH3000, GH Superchip 72C 3GHz, NVIDIA GH200 Superchip, Quad-Rail NVIDIA InfiniBand NDR200, RedHat Enterprise Linux, EVIDEN EuroHPC/FZJ Germany	4,801,344	1,000.00	1,226.28	15,794
5	Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	

TOP500 – and where is Denmark?

- ❑ two entries on the TOP500 list as of June 2017:

Rank	System	Cores	Rmax [TFlop/s]	Rpeak [TFlop/s]	Power [kW]
441	Vestas1 - Lenovo NeXtScale nx360M5, Xeon E5-2680V3/Xeon E5-2680v4 14C 2.4GHz, Infiniband FDR , Lenovo Vestas Wind Systems A/S Denmark	16,848	481.8	522.5	
459	Abacus 2.0 - Lenovo NeXtScale nx360M5, Xeon E5-2680v3 12C 2.5GHz, Infiniband FDR, NVIDIA Tesla K40 , Lenovo University of Southern Denmark Denmark	17,928	462.4	836.6	187.5

- ❑ since then, DK has not been on the list any longer!
- ❑ there were probably more powerful installations in DK, but they did not “want” to be on the list
- ❑ Denmark owns a share of LUMI (no. 8 in 2024 list)

TOP500 – and where is Denmark?

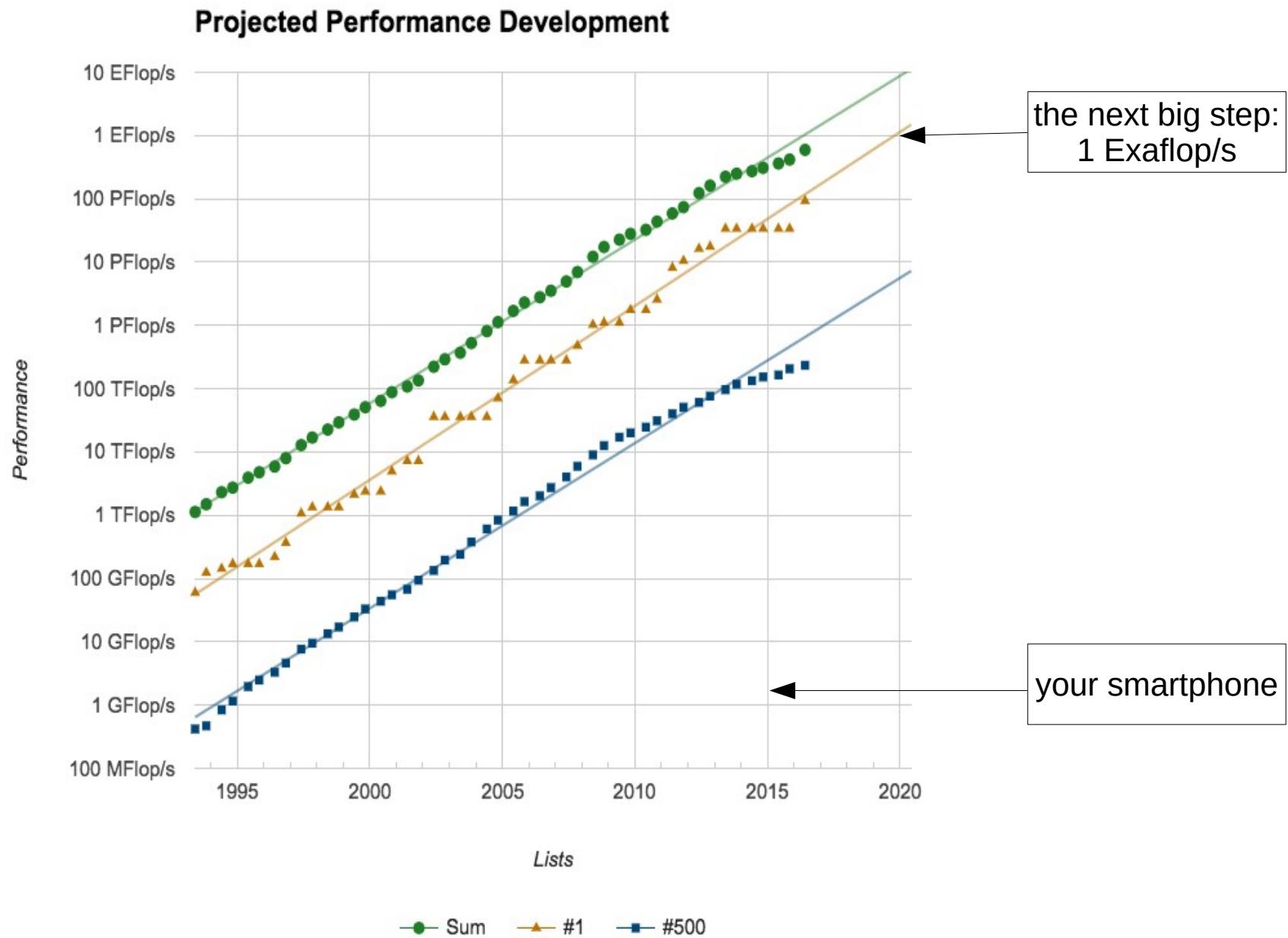
- ❑ The good news Nov 2024: Denmark is back!

1 entries found.

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
21	Gefion - NVIDIA DGX H100, Xeon Platinum 8480C 56C 2GHz, NVIDIA H100 SXM5 80GB, Octo-rail NVIDIA HDR100 Infiniband, Red Hat Enterprise Linux, Nvidia Danish Centre for AI Innovation Denmark	223,088	66.59	100.63	1,753

- ❑ 191 Nvidia DGX H100 systems
- ❑ 1528 H100 GPUs
- ❑ access through DCAI <https://dcai.dk/>
- ❑ November 2025: rank 34

TOP500 – history and outlook (2017)



TOP500 – HPC's Formula 1

- Some remarks:
 - not always applicable to 'real world' problems
 - (sometimes) difficult to program
 - huge installations → power issues
 - The TOP500 no. 1 (El Capitan) uses about 29-30 MW
- An alternative list – Green500:
 - <https://top500.org/lists/green500/>
 - measures the power efficiency: Mflop/s / W
 - number 1 on the Green500 list is number 222 on the TOP500 (TOP500 no. 1 → Green500 no. 18)

TOP500 – the Exaflop/s challenge

- ❑ first projections said, that the world will see the first Exaflop/s machine around 2018
- ❑ since then, it has been postponed several times – the latest prediction said ~2022
- ❑ Frontier entered in June 2022 with 1.1 Exaflop/s
- ❑ Challenges:
 - ❑ power consumption (goal: max 20MW!)
 - ❑ memory technologies
 - ❑ ...
- ❑ but there are always surprises, e.g. new CPU designs and/or other technologies, like GPUs

The HPC landscape is changing ...

HPC methods are penetrating all areas of computing

- ❑ embedded systems based on multi-core
- ❑ use of GPUs as accelerators on desktop and laptop systems
 - ❑ mainly driven by AI and Machine Learning
- ❑ “Big Data” - HPC methods for high-performance data analytics and visualization

The HPC landscape is changing ...

There are currently discussions about a new and updated benchmark for the TOP500 list

- ❑ the current HPC Linpack is not very realistic (dense matrices)
- ❑ add more realistic scenarios, e.g. sparse matrix calculations
- ❑ new list: TOP500 HPCG list
 - ❑ High-Performance Conjugate Gradient
 - ❑ no. 1: Fugaku (Japan)
- ❑ add power consumption or a power envelope
- ❑ more ...

High-Performance Computing



