Introduction to High-Performance Computing

Session 01

Organization

Introduction to HPC

but before we begin…

Course Organisation

Contact Information

Stefan Harfst (Scientific Computing)

email: stefan.harfst@uni-oldenburg.de

office: W03 1-139

web: http://www.uni-oldenburg.de/fk5/wr/

Wilke Trei (ForWind)

email: wilke.trei@forwind.de

office: W33 3-323

Course Organisation

Monday	Wednesday	Thursday
10:15 – 11:15 (SH) Introduction to HPC	*10:15 – 11:15 (WT)* Introduction to Parallel Programming	*10:15 – 11:15 (SH)* Introduction to Matlab Distributed Computing (MDCS)
11:30 – 13:00 (WT) Basic Cluster Usage	*11:30 – 13:00 (WT)* Introduction to OpenMp	*11:30 – 13:00 (SH)* Parallel Programming in Matlab
13:45 – 15:15 (SH) HPC Environment	*13:45 – 15:15 (SH)* Introduction to OpenMP	*13:45 – 15:15 (SH)* Parallel Programming in Matlab
15:30 – 17:00 (SH) Advanced Cluster Usage	*15:30 – 17:00 (SH)* Performance Optimization	

- times are only approximate

- lectures and practical parts
- slides and files for practical parts will be available in the Wiki

Hands-on Sessions

- we will be using the local HPC clusters CARL and EDDY
- if you don‘t have an account yet go to http://www.uni-oldenburg.de/fk5/wr/
  - there are useful links to request an account and also to the HPC wiki which is the central resource for information
  - if you are not a member of a research group you can choose “student” (can be changed later if needed)

Introduction HPC

What is High-Performance Computing?

- possible answer:
  - if we ask google the first link says

> “High Performance Computing most generally refers to the practice of aggregating computing power in a way that delivers much higher performance than one could get out of a typical desktop computer or workstation in order to solve large problems in science, engineering, or business.”

(http://insidehpc.com/hpc-basic-training/what-is-hpc/)

  - in short: use a big computer (HPC cluster) to solve a big problem

What is High-Performance Computing?

IBM's Blue Gene/P supercomputer at Argonne National Laboratory
taken from https://en.wikipedia.org/wiki/Supercomputer

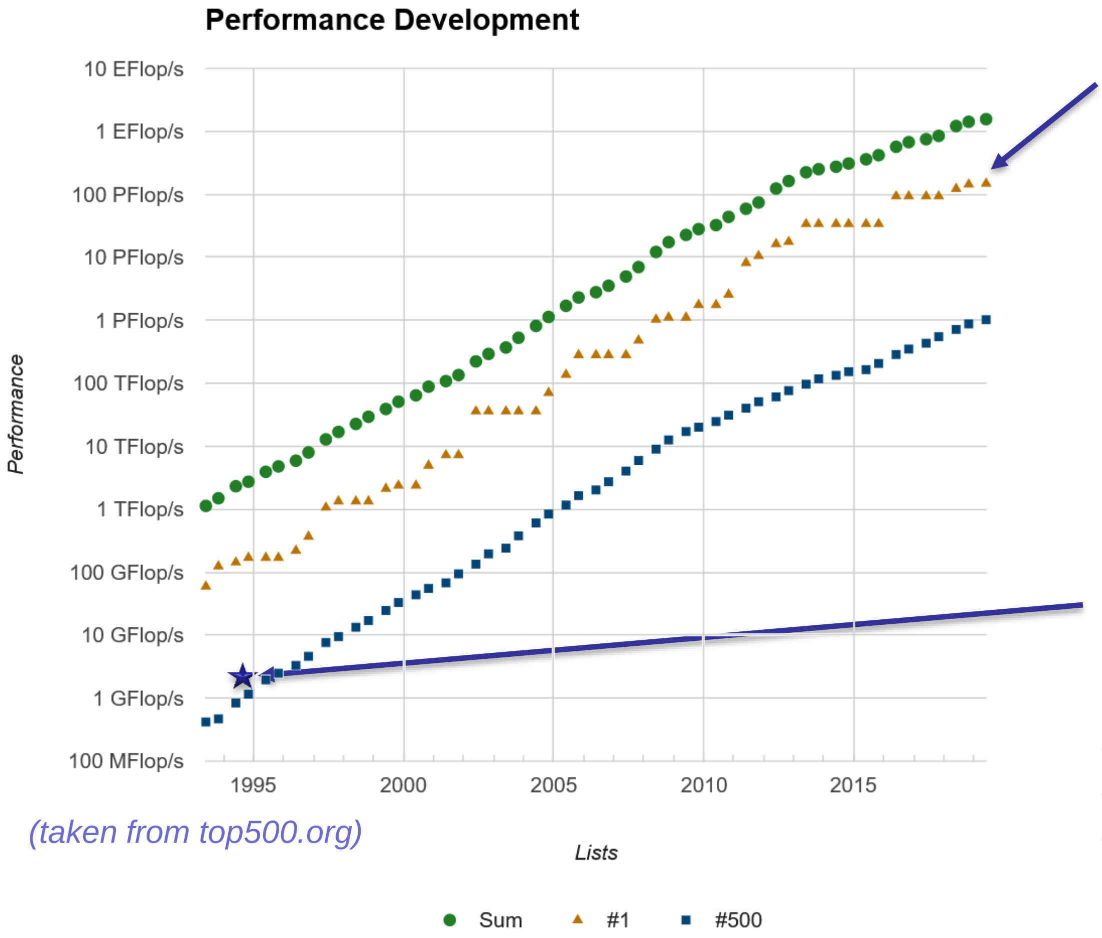
What is High-Performance Computing?

- another answer:
  - „computing at the bottleneck“ (from G. Hager @ RRZE, see https://www.rrze.fau.de/dienste/arbeiten-rechnen/hpc/HPC4SE/)
  - what is the optimal performance I can achieve on a single core or a multi-core processor?
  - what is the limiting factor? e.g. CPU-speed vs. memory access

High-Performance Computing (HPC)

- why HPC?
  - enables us to solve computationally intensive problems
- requirements
  - understanding of the used computational architecture
  - identification of computing bottlenecks
  - parallelization and optimization of algorithms
- typical HPC resources
  - HPC Cluster (e.g. CARL and EDDY)
  - Grid-Computing (distributed resources, e.g. SETI@home)
  - GPUs

The Fastest Computers on Earth



(taken from top500.org)

#1 June 2019: Summit
(Oak Ridge National Lab, United States)
4608 nodes (2x IBM Power9 22c +
6x Nvidia V100)

Expected measured Peak Performance

Current Tablet ≈ 1.5 Gflop/s

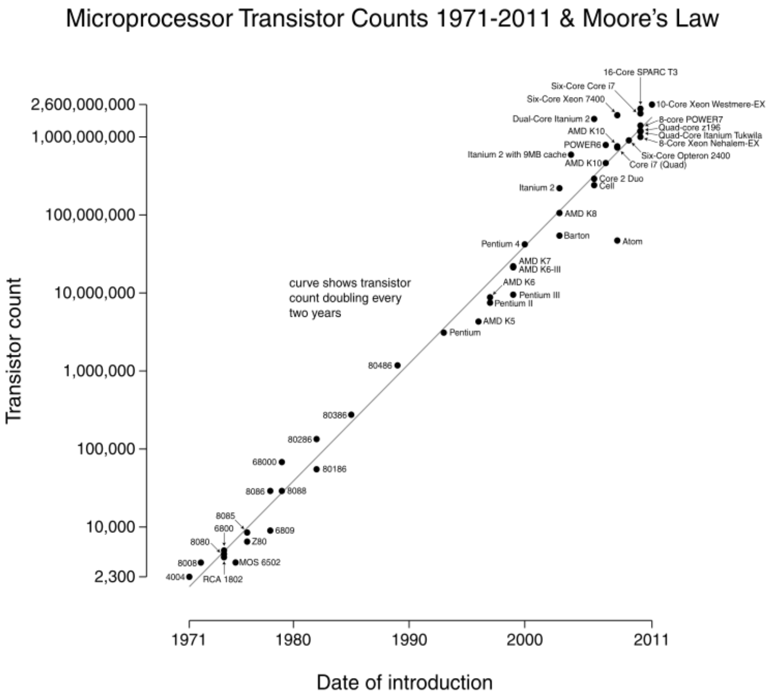
would have made it into the list in 1994

http://bits.blogs.nytimes.com/2011/05/09/the-ipad-in-your-hand-as-fast-as-a-supercomputer-of-yore/?_r=1

Moore’s Law

- the number of transistors in a dense integrated circuit doubles every 12 to 24 months
- often interpreted as computing speed doubles every ~18 months
- empirical law and self-fulfilling prophecy
- Moore’s Law must eventually break down

there are indications that we are already seeing the end of Moore’s Law



(taken from Wikipedia)

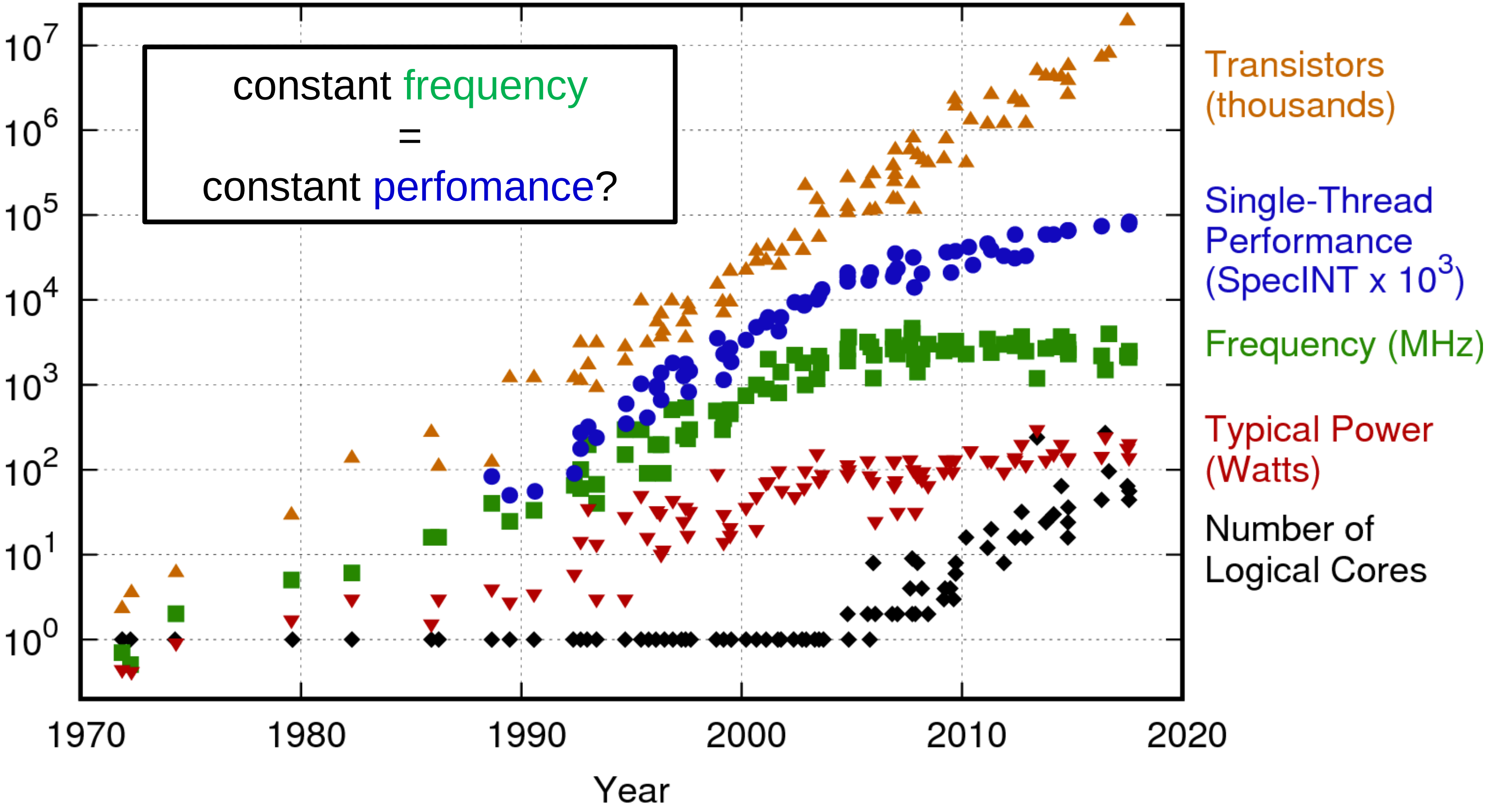
The End of Moore‘s Law?

- see e.g. http://www.nature.com/news/the-chips-are-down-for-moore-s-law-1.19338
- it is probably not economic to shrink below a few nm
- what comes next?
  - not known yet but it will be interesting
  - one possible way is to go 3d, already seen in memory
  - quantum computer? see https://singularityhub.com/2019/02/26/quantum-computing-now-and-in-the-not-too-distant-future or https://www.research.ibm.com/ibm-q/learn/what-is-quantum-computing/

Parallelization

- parallelization is needed if the computational power of a single computer is not enough
  - distribution of data and/or work among several computers
  - different strategies depending on the problem at hand and also the available parallel architecture
  - parallel computing is very important today and will be more important in the future
  - modern PCs have multiple core CPUs → can be utilized by parallel applications

42 Years of Microprocessor Trend Data



constant frequency
=
constant perfomance?

https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

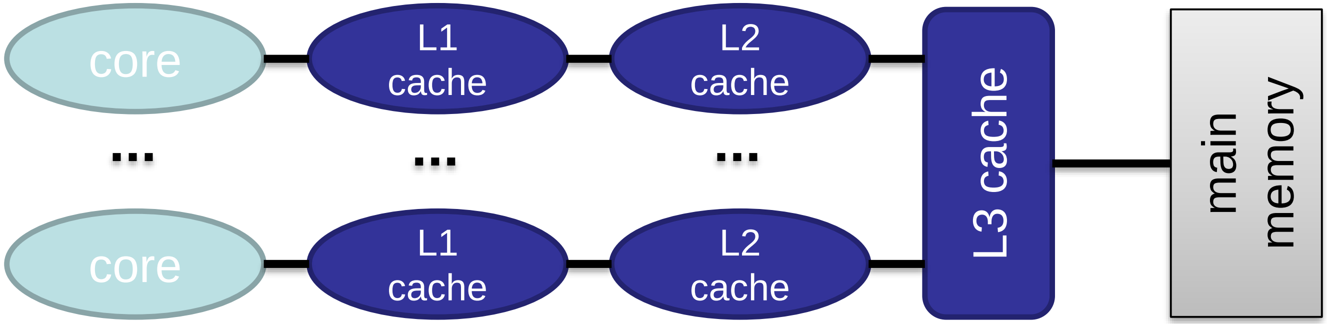
Parallel Architectures

- the knowledge of the underlying parallel architecture is important
  - allows to write more efficient code
  - avoid bottlenecks
  - understand the limiting factors for performance

- typical HPC cluster (e.g. CARL and EDDY)
  - multi-core CPU
  - multi-socket node
  - many nodes with node interconnect

Parallel Architectures

- multi-core CPU
  - contains several cores (currently up to 22 or so)
  - cores connected to caches for fast memory access, low latency (order of ten times faster than direct memory access)
  - cache coherence



Broadwell E5-2650 | 12 cores, 2.2GHz | 64kB | 256kB | 30MB

Parallel Architectures

- how to **get** information about node architecture?
  - tools exist, e.g. hardware locality (hwloc) software, see example:

```
$ module load hwloc
$ hwloc-ls
Machine (255GB total)
  NUMANode L#0 (P#0 127GB)
    Package L#0 + L3 L#0 (30MB)
      L2 L#0 (256KB) + L1d L#0 (32KB) + L1i L#0 (32KB) + Core L#0
      L2 L#1 (256KB) + L1d L#1 (32KB) + L1i L#1 (32KB) + Core L#1
      ...
```

- how to **use** information about node architecture?
  - optimize cache usage
  - make decision about binding processes to cores

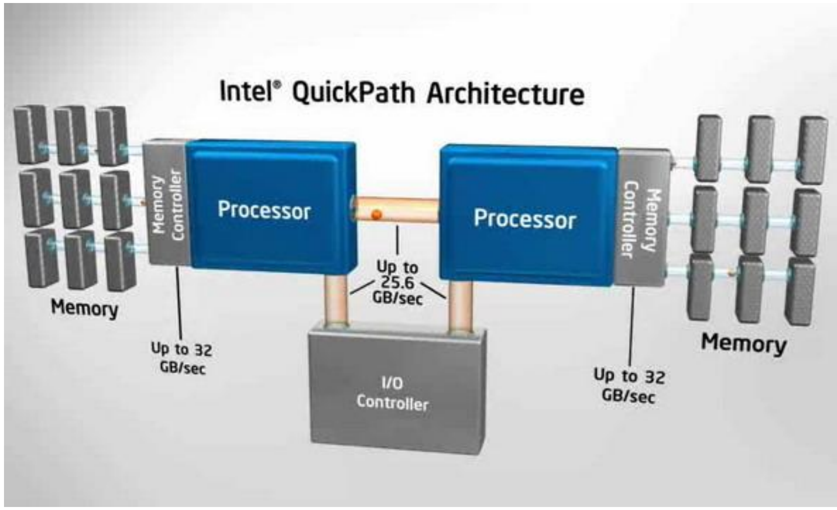
not trivial to do, best to rely on tested libraries and default settings, start to care when your application is slower than expected

Parallel Architectures

- multi-core CPU
  - number of cores increases, drives the performance increase
  - uniform memory access (UMA)
  - each core can execute single instructions on multiple data (SIMD), e.g. SSE, AVX, AVX2
  - clock frequency stagnates or even decreases (thermal effects)
  - computations can be limited by memory access (band width) or by the number of instructions per cycle

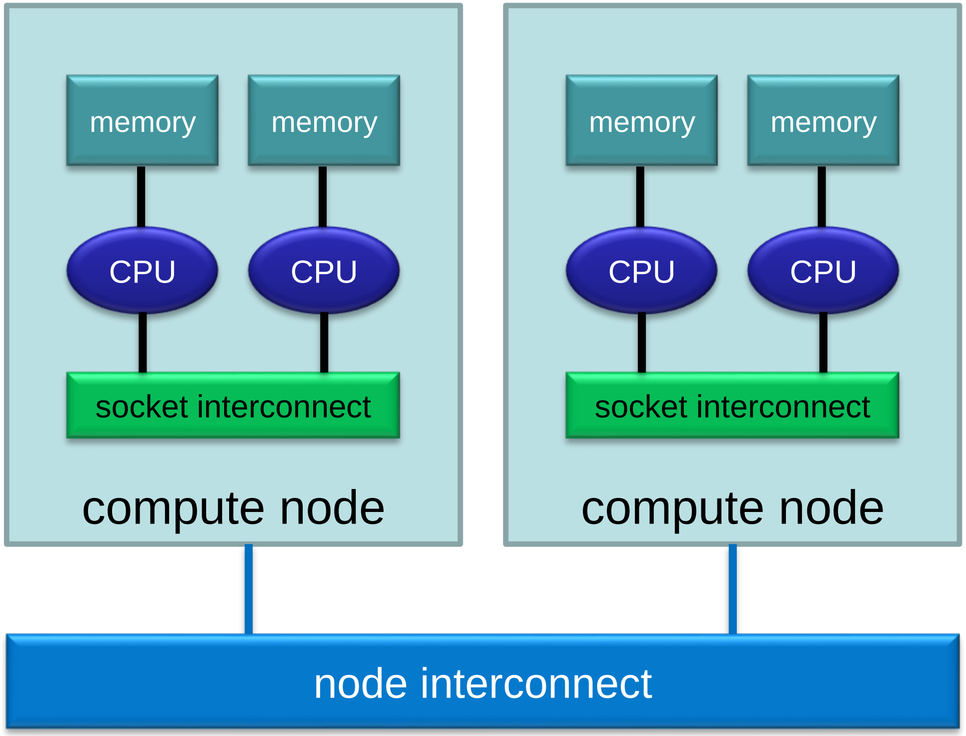
Parallel Architectures

- multi-socket node
  - several CPUs on a single main board
  - each CPU has access to its own main memory and cache levels
  - different, non-uniform memory access (NUMA)
  - with cache coherence → ccNUMA

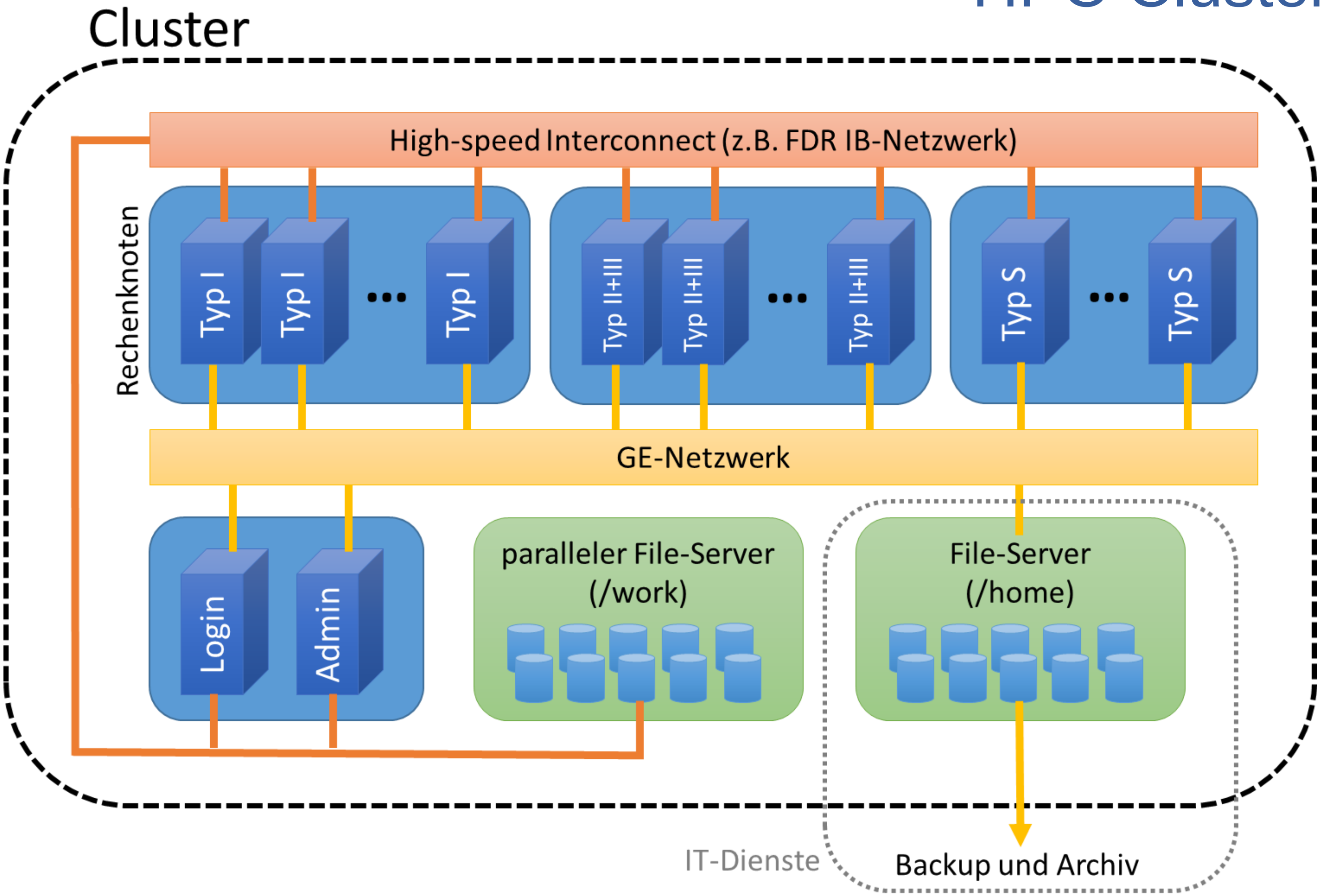


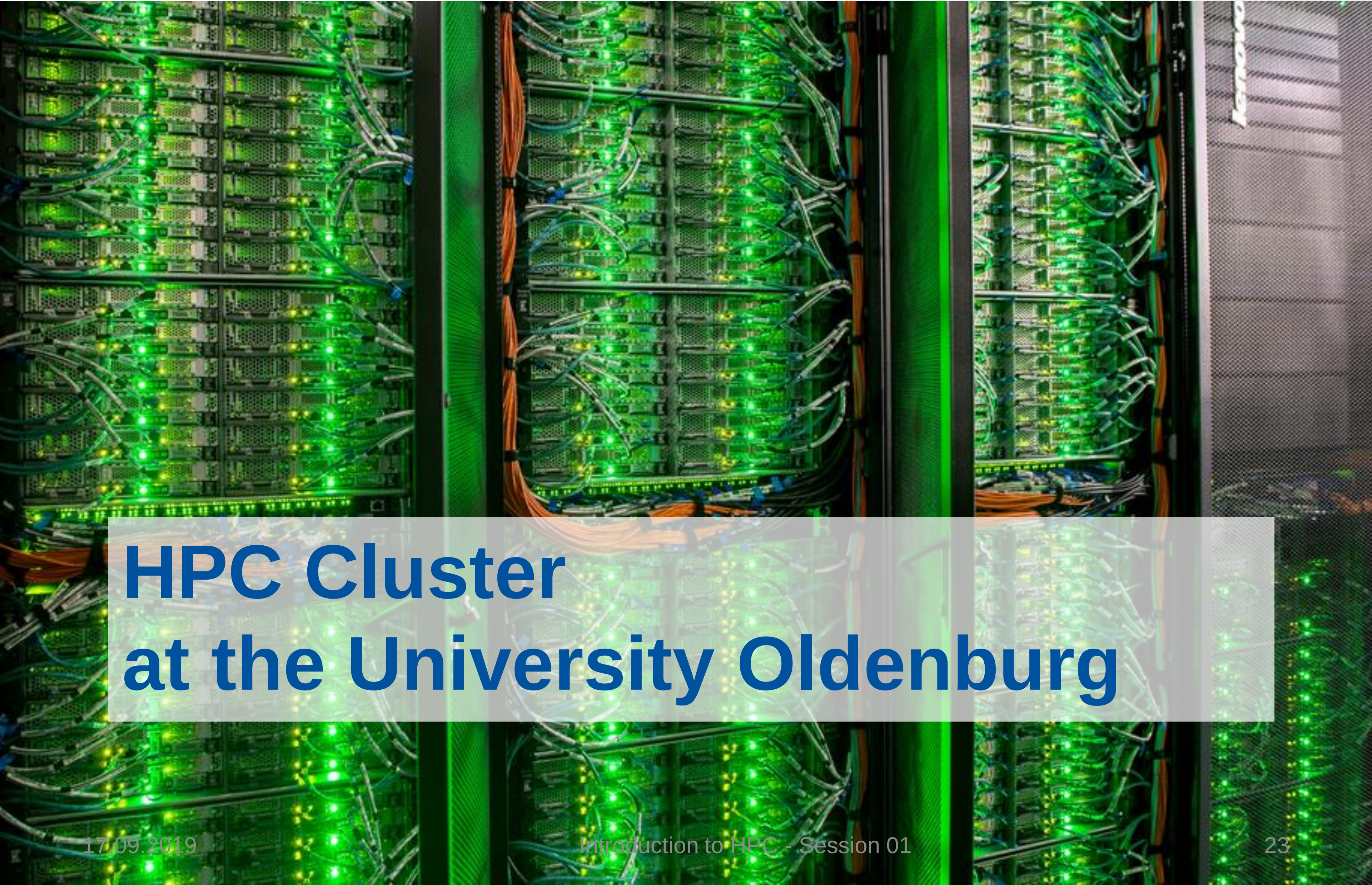
Parallel Architectures

- multi-computer
  - many (ccNUMA) nodes
  - distributed memory
  - node interconnect



HPC Cluster


Cluster
High-speed Interconnect (z.B. FDR IB-Netzwerk)
Rechenknoten
Typ I
Typ I
...
Typ I
Typ II+III
Typ II+III
...
Typ II+III
Typ S
...
Typ S
GE-Netzwerk
Login
Admin
paralleler File-Server (/work)
File-Server (/home)
IT-Dienste
Backup und Archiv



HPC Cluster
at the University Oldenburg

Delivery Day

- the new hardware was delivered Aug 22nd, 2016

http://www.uni-oldenburg.de/fk5/wr/aktuelles/artikel/art/neue-hochleistungsrechner-fuer-die-universitaet-oldenburg-2380/

Overview New Hardware

- CARL
  - multi-purpose cluster as a basic computing resource
  - funded by the University/MWK and the DFG under grant number INST 184/157-1 FUGG (Forschungsgroßgerät nach Art. 91b GG)
- EDDY
  - CFD cluster for wind energy research
  - funded by the BMWi under grant number 0324005
- used as a shared HPC cluster
  - common infrastructure is shared (e.g. file systems, network)
  - shared administration

http://wiki.hpcuser.uni-oldenburg.de/index.php?title=Acknowledging_the_HPC_facilities_2016

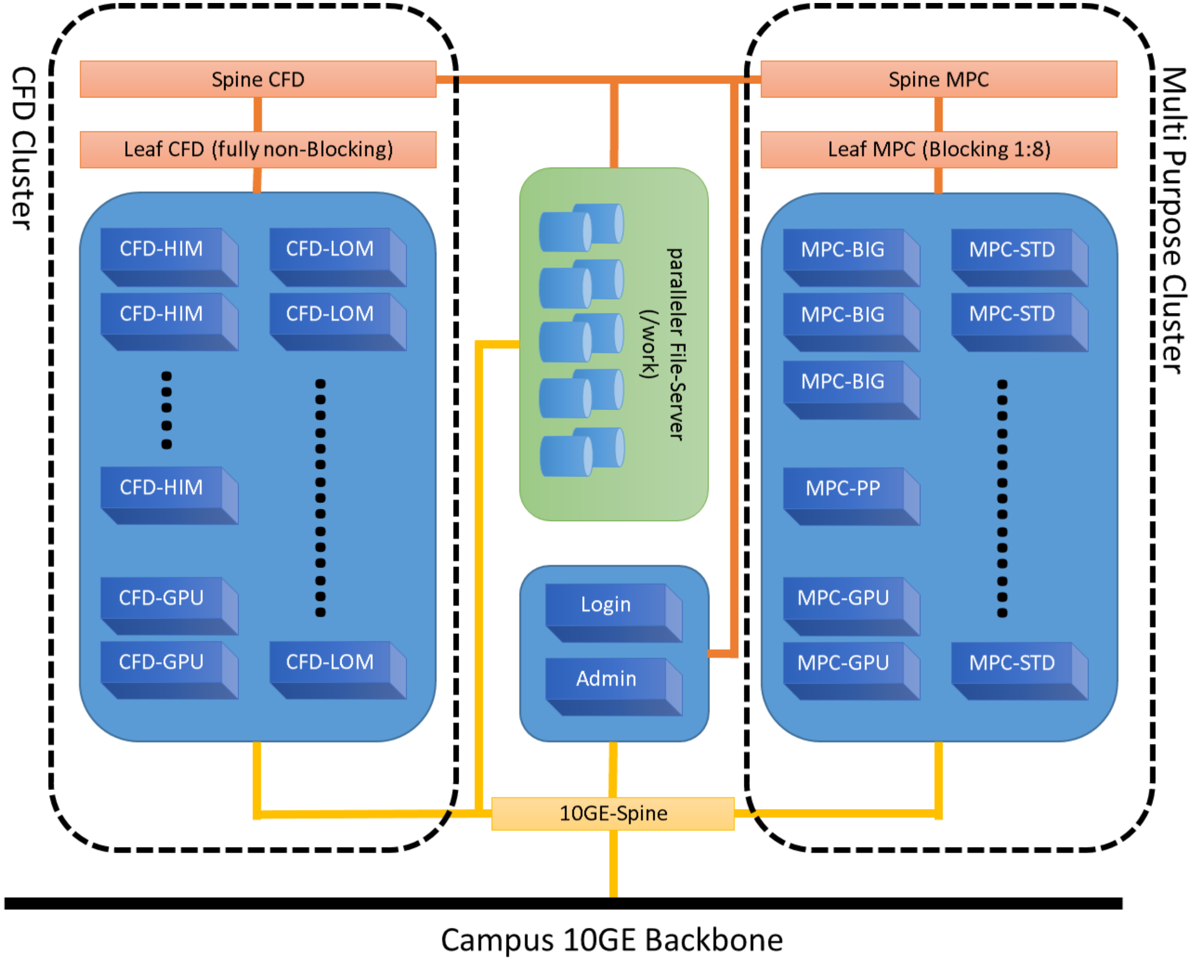
HPC Facilities @ University Oldenburg

- shared HPC cluster CARL/EDDY
  - close to 600 compute nodes
  - 4 login and 2 administration nodes
  - Infiniband FDR interconnect for parallel computing
  - 10/1GE network
  - parallel file system (GPFS) with 900TB capacity
  - NFS mounted central storage

  - Linux (RHEL) as OS
  - many scientific applications and libraries available
  - Job Scheduler (SLURM)

https://wiki.hpcuser.uni-oldenburg.de/index.php?title=HPC_Facilities_of_the_University_of_Oldenburg_2016

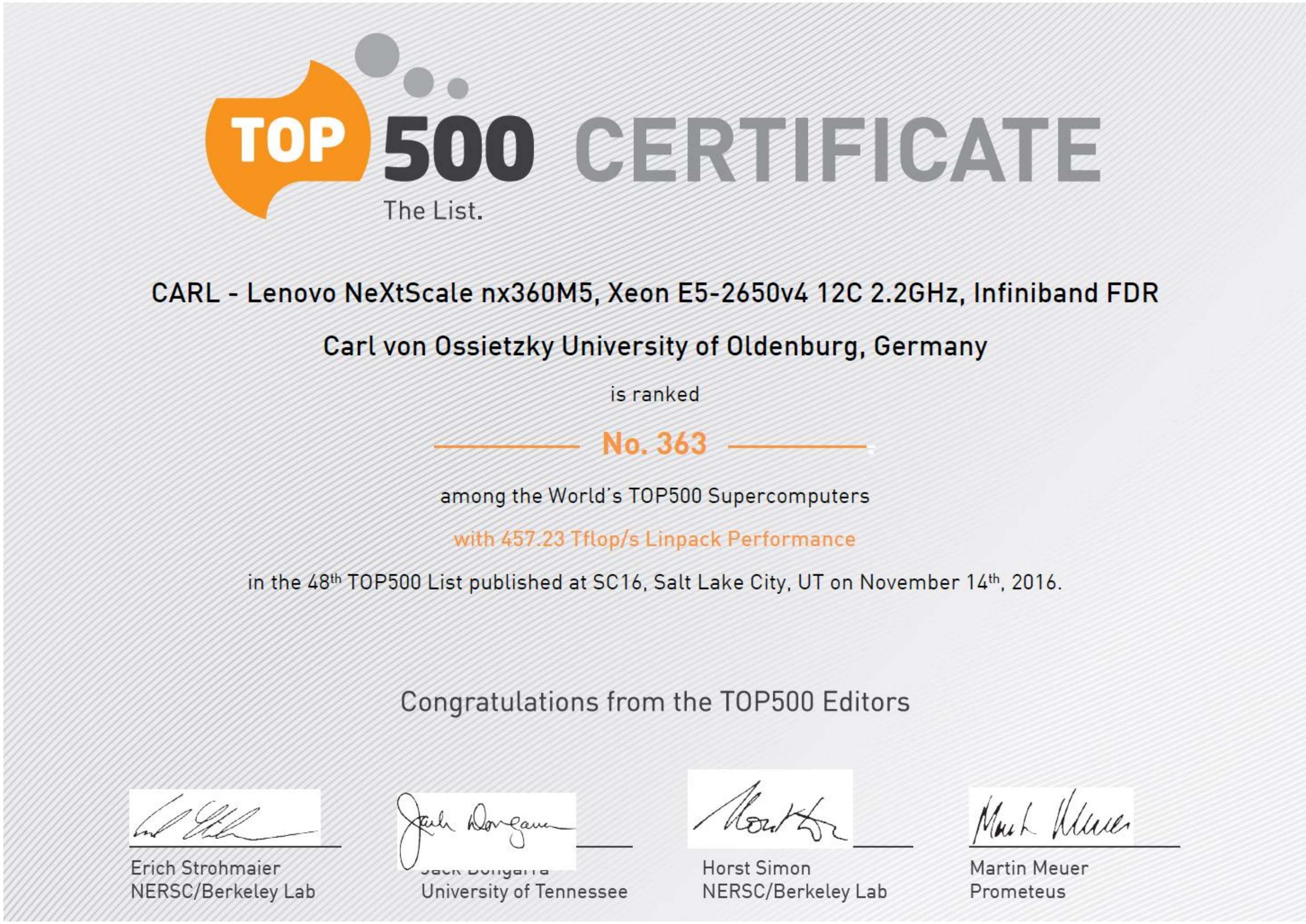
Schematic View of HPC Cluster



Summary CARL & EDDY

Feature	CARL	EDDY	Total
Nodes	327	244	571
Cores	7.640	5.856	13.496
RAM	77 TB	21 TB	98 TB
GPFS	450 TB	450 TB	900 TB
local disks	360 TB	-	360 TB
Rpeak (nominal)	271 Tflop/s	201 Tflop/s	482 Tflop/s
Rpeak (AVX2)	221 Tflop/s	164 Tflop/s	385 Tflop/s
Rmax			457.2 Tflop/s

Rank 363 in Top500 https://www.top500.org/system/178942



TOP500 CERTIFICATE

The List.

CARL - Lenovo NeXtScale nx360M5, Xeon E5-2650v4 12C 2.2GHz, Infiniband FDR

Carl von Ossietzky University of Oldenburg, Germany

is ranked

No. 363

among the World´s TOP500 Supercomputers

with 457.23 Tflop/s Linpack Performance

in the 48th TOP500 List published at SC16, Salt Lake City, UT on November 14th, 2016.

Congratulations from the TOP500 Editors

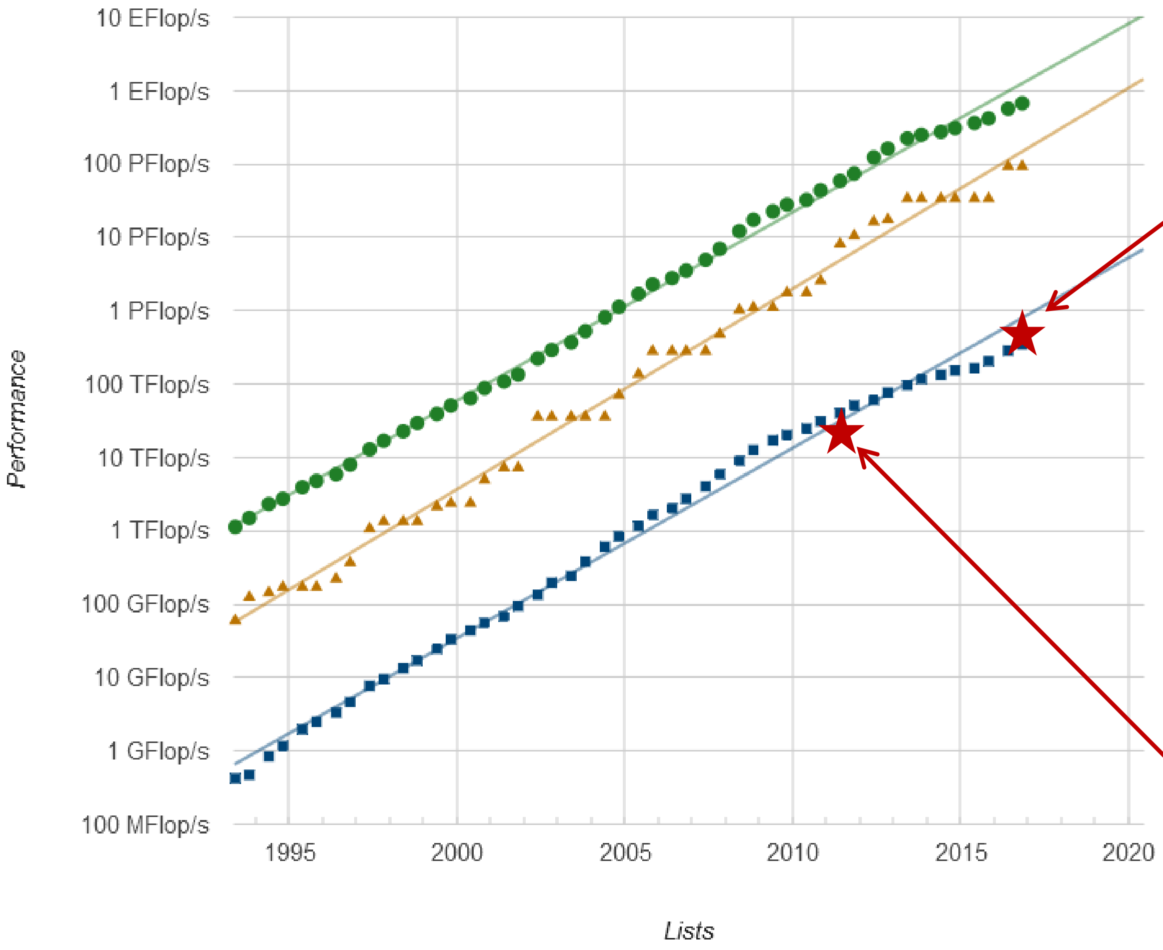
Erich Strohmaier
NERSC/Berkeley Lab

Jack Dongarra
University of Tennessee

Horst Simon
NERSC/Berkeley Lab

Martin Meuer
Prometeus

Top500 Performance Development



CARL/EDDY

- Rmax 457.2 Tflop/s
- Rank 363
- 571 nodes
- 13,500 cores
- 100 TB RAM

17x Rmax
1.6x nodes
3.3x cores
10x RAM

HERO/FLOW

- Rmax 27.2 Tflop/s
- not ranked
- 350 nodes
- 4,000 cores
- 10 TB RAM

Racks in Server Room

Login to the HPC Cluster

http://wiki.hpcuser.uni-oldenburg.de/index.php?title=Login

- Linux
  - use ssh as before with **carl** or **eddy** as login nodes

    ```
    ssh -X abcd1234@carl.hpc.uni-oldenburg.de
    ```
- Windows
  - use MobaXterm (recommended) or PuTTY

- login host names

  ```
  hpcl00[1-4].hpc.uni-oldenburg.de
  ```
  - can be used instead of **carl** or **eddy** (for login to specific node)
  - no difference between **carl** and **eddy** as login

- from outside of the campus network use VPN connection
  - see instructions at http://www.itdienste.uni-oldenburg.de/21240.html