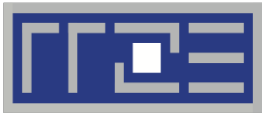


HPC Café – today @ 4 pm: AI-assisted research at FAU

- 4:00 - 4:15 **General intro, announcements, Q&A**
- 4:15 - 4:30 **Peter Uhrig: "HPC workflows for big data and machine learning applications"**
- 4:30 - 4:45 **Benedikt Lorch: "Deep learning in the HPC environment Computing demands for research in multimedia forensics"**
- 4:45 - 5:00 **Harald Köstler: "Deep learning for Computational Fluid Dynamics"**
- 5:00 - 5:15 **Thorsten Glüsenkamp: "AI research in astroparticle physics at ECAP"**
- 5:15 pm Discussion (open end)

Tomorrow 4 pm: HPC intro - how to access the HPC systems, start jobs, etc.



Erlangen Regional
Computing Center



Changes to expect for TinyGPU in 2021 and other news

Dr. Thomas Zeiser



Recent / coming hardware news

- 31.12.2020: tg00x with **Geforce GTX980** have been **powered off**
 - ➔ no nodes anymore with SSE2 only
 - ➔ all remaining nodes have a scratch SSD
- Early 2021: **5 nodes with 4x Nvidia A100** each to come
 - 32 AMD host cores and 128 GB host memory available per A100
- Hopefully by the end of Q1/2021:
7 nodes with 8x Nvidia Geforce RTX3080 each to come
 - still 4 Intel host cores per RTX3080 only
- **5 groups spent ~310.000 EUR** to make the purchase of the A100 and RTX3080 nodes possible ➔ special access restrictions apply
- *TinyFat also has been extended by 36 nodes with AMD 64 core processors and 512 GB main memory each (~200k EUR spent by 3 groups)*

Operation model of the new GPU and FAT nodes (I)

- **Ubuntu 20.04 LTS** (instead of Ubuntu 18.04 LTS)
 - there is currently no login node with 20.04 yet
 - request an (interactive) compute node to compile on 20.04
- **Only recent versions of software** are installed on a **separate /apps**
 - e.g. only CUDA/11.2, or Intel 20 compilers
 - much of the **existing software** will fail to run and thus **needs to be recompiled**
 - we might provide a Singularity container with the Ubuntu 18.04 environment to run existing binaries for a transition period
 - output of the *module* command is slightly different due to a more recent version
- **Slurm as batch system** instead of Torque
 - documentation and sample scripts will be provided in the coming weeks
 - Woody frontends can be used to submit Slurm jobs to TinyGPU/TinyFAT

Operation model of the new GPU and FAT nodes (II)

- **Owners will have the option of a QOS priority boost in Slurm**
 - nodes are no longer reserved for owner groups
 - longer job run times (at least 48h instead of 24h)
 - priority jobs can **preempt** regular jobs, i.e. regular jobs will be killed if owners claim their hardware
 - make your jobs aware of early termination
 - use application-level checkpointing
 - trap signals to react on preemption
 - check for error conditions to avoid chain jobs running amok, i.e. with resubmit themselves immediately without doing work as the command fails because of corrupt input, etc. → only resubmit if runtime was at least several minutes
 - but owners can only claim as much hardware as they financed with the **QOS priority boost**; owners (and everyone else) can run on more hardware without QOS priority boost – but all these jobs are subject to preemption

Outlook

- Documentation is soon moving from <https://anleitungen.rrze.fau.de/hpc> to <https://hpc.fau.de/systems-services/systems-documentation-instructions/>
- Remaining parts of TinyGPU, TinyFAT as well as Woody will be upgraded to Ubuntu 20.04 and Slurm later in the year
 - probably in summer?
 - once Woody switched to Slurm, more single-core jobs will be possible; no need to allocate full nodes anymore
- TinyEth may be retired later in the year
 - at latest when new throughput resources are financed by a physics group

Final remarks

- Be as generic as possible when requesting resources
 - We often see that many nodes are unused while jobs are waiting in the queue because they request (too) specific node types
 - A V100 may be faster than a GTX1080 but if you have to wait for days for your **:v100** job to start it may not pay out → do not request a specific GPU if possible
 - Similarly on Woody; there is little use of specifying **:sl32g** / **:kl32g**; use **:any32g** if you depend on 32 GB main memory (or no property otherwise)
- In Emmy there are 4 nodes with one Tesla V100 (16 GB) in addition to the Tesla K20 nodes.
 - **:anygpu** (13 nodes), **:k20m** (9 nodes), **:k20m1x** (4 nodes), **:k20m2x** (5 nodes), **:v100** (4 nodes)