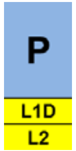
High Performance Computing in a Nutshell

HPC Services, RRZE / NHR@FAU

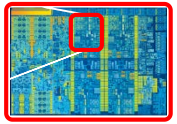
HPC systems at RRZE

https://hpc.fau.de/systems-services/systems-documentation-instructions/

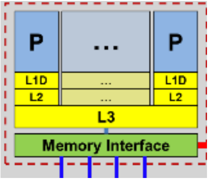
Parallel computing hardware terminology

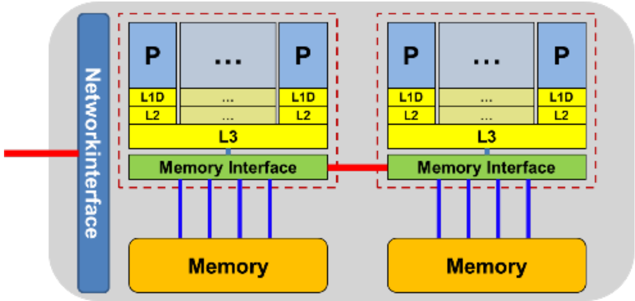


core



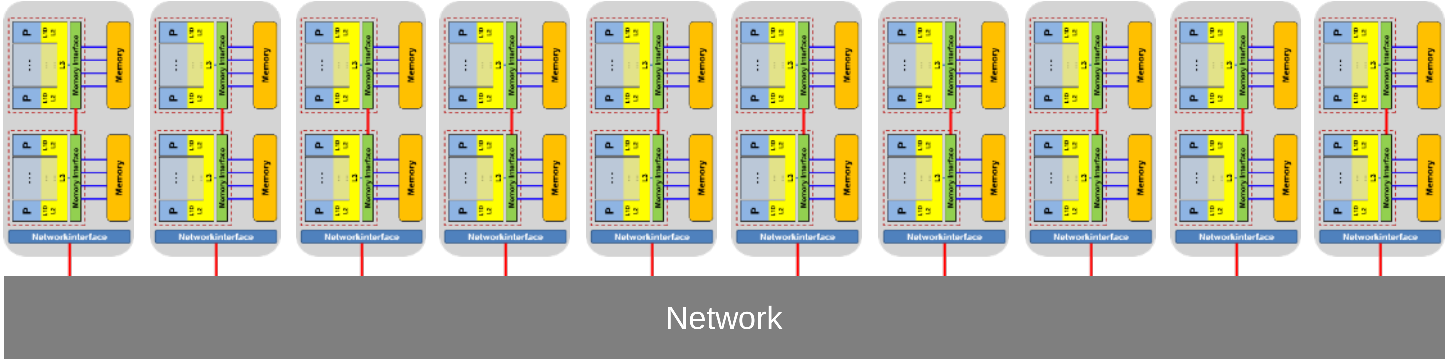
chip/socket
“CPU”





shared-memory compute node

distributed-memory cluster

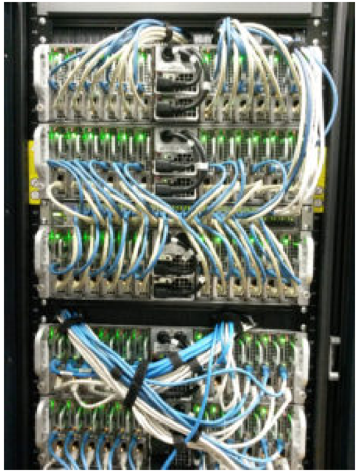


RRZE “Woody” cluster + “TinyEth”

main workhorse for throughput and single-node jobs

Woody:

- all 246 nodes with 4 cores and high clock frequency (3.5/3.7 GHz) Intel Xeon E3-1240 v? processors
  - 70x Intel Haswell, 8 GB RAM
  - 64x Intel Skylake, 32 GB RAM
  - 112x Intel Kaby Lake, 32 GB RAM
- at least 900 GB local HDD/SSD
- and Gbit only



TinyEth:

- 20 nodes (480 cores)
  - 12 cores @ 2.66 GHz
  - 48 GB RAM
  - 30/190/420 GB local HDD
- Single cores can be requested

RRZE “Emmy” cluster

main workhorse for parallel jobs

- 543 compute nodes (10.880 cores)
  - 2 Intel Xeon E5-2660v2 (Ivy Bridge) 2.2 GHz (10 cores)
  - 20 cores/node + SMT cores
- 64 GB main memory per node
- No local disks
- Full QDR Infiniband fat tree network: 40 GBit/s and < 2 µs latency

NEC
Amalie Emmy Noether
Emmy Noether lieferte fundamentale Beiträge zur abstrakten Algebra und zur theoretischen Physik. Das nach ihr benannte Noether-Theorem beschreibt die Verbindung zwischen Symmetrien physikalischer Naturgesetze und Erhaltungsgrößen.
Parallelrechner "Emmy"
Installation Q4/2013
560 Rechenknoten
11200 Rechenkerne "Intel Ivy Bridge"
16 Xeon Phi Koprozessoren
16 Nvidia K20 GPGPUs
234 TFlop/s Spitzenrechenleistung
36 TByte Hauptspeicher
QDR-InfiniBand-Netzwerk
450 TByte Paralleles Filesystem
FAU
Amalie Emmy Noether

RRZE “Meggie” cluster

for scalable parallel jobs – prior account activation required

- 728 Compute nodes (14.560 cores)
  - 2 Intel Xeon E5-2630 v4 (Broadwell) 2.2 GHz (10 cores)
  - 20 cores/node
  - 64 GB main memory
- No local disks
- Intel OmniPath network: Up to 100 Gbit/s

- Peak Performance:
  R_{peak} = 0.5 PF/s

RRZE “TinyGPU” cluster

for GPU workloads – not all nodes always generally available

- 7 nodes with 2x “Broadwell” @2.2 GHz, 64 GB RAM, 980 GB SSD, 4x GTX1080
- 10 nodes with 2x “Broadwell” @2.2 GHz, 64 GB RAM, 980 GB SSD, 4x GTX1080Ti
- 12 nodes with 2x “Skylake” @ 3.2 GHz, 96 GB RAM, 1.8 TB SSD, 4x RTX 2080Ti
- 4 nodes with 2x “Skylake” @3.2 GHz, 96 GB RAM, 2.9 TB SSD, 4x Tesla V100
- (5 nodes with 2x AMD Rome 7662 @2.0 GHz, 512 GB RAM, 5.8 TB SSD, 4x Volta A100)

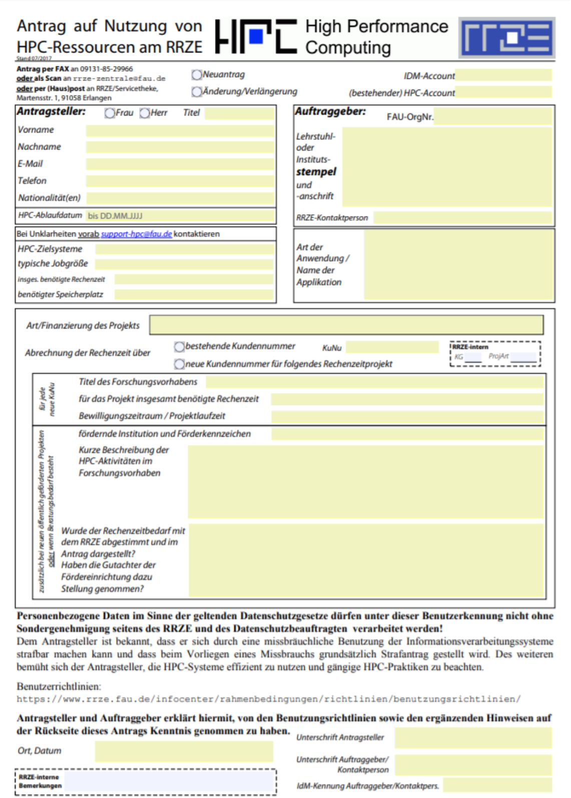
What is each system good for?

Cluster	#nodes	Appl.	Parallel FS	Local HDD	Description
Meggie	728	massively parallel	Yes	No	Newest RRZE cluster, highly parallel workloads. Access restricted.
Emmy	560	massively parallel	Yes	No	Current main cluster for parallel jobs
Woody	246	serial, single-node, throughput	No	Yes, some w/ SSD	High clock speed single-socket nodes for serial throughput
TinyEth	20	single-node, throughput	No	Yes	Throughput workloads
TinyGPU	38	GPGPU	No	Yes, all w/ SSD	Different types of Nvidia GPGPUs; Access restrictions and throttling policies may apply
TinyFat	46	Large memory	No	Yes, all w/ SSD	256-512 GB memory per node. Access restrictions may apply.

Accessing HPC systems at RRZE

HPC account

- You need a separate account (not your IdM account)
- HPC account application form
- Account can access all HPC systems at RRZE!
- Ask your local RRZE contact person for help
- If you change your affiliation, you need a new HPC account. Data migration may be required



Antrag auf Nutzung von HPC-Ressourcen am RRZE

High Performance Computing

Stand 07/2017

Antrag per FAX an 09131-85-29966
oder als Scan an `rrze-zentrale@fau.de`
oder per (Haus)post an RRZE/Servicetheke,
Martensstr. 1, 91058 Erlangen

○ *Neuantrag*
○ *Änderung/Verlängerung*

IDM-Account
(bestehender) HPC-Account

Antragsteller: ○ *Frau* ○ *Herr* *Titel*

Vorname
Nachname
E-Mail
Telefon
Nationalität(en)
HPC-Ablaufdatum bis DD.MM.JJJJ

Auftraggeber: FAU-OrgNr.

Lehrstuhl- oder Instituts- ***stempel*** *und -anschrift*

RRZE-Kontaktperson

Bei Unklarheiten vorab support-hpc@fau.de kontaktieren

HPC-Zielsysteme
typische Jobgröße
insges. benötigte Rechenzeit
benötigter Speicherplatz

Art der Anwendung / Name der Applikation

Art/Finanzierung des Projekts

Abrechnung der Rechenzeit über ○ *bestehende Kundennummer* *KuNu*
○ *neue Kundennummer für folgendes Rechenzeitprojekt*

RRZE-intern *KG* *ProjArt*

für jede neue KuNu
Titel des Forschungsvorhabens
für das Projekt insgesamt benötigte Rechenzeit
Bewilligungszeitraum / Projektlaufzeit

zusätzlich bei neuen öffentlich geförderten Projekten oder wenn Beratungsbedarf besteht
fördernde Institution und Förderkennzeichen
Kurze Beschreibung der HPC-Aktivitäten im Forschungsvorhaben
Wurde der Rechenzeitbedarf mit dem RRZE abgestimmt und im Antrag dargestellt? Haben die Gutachter der Fördereinrichtung dazu Stellung genommen?

Personenbezogene Daten im Sinne der geltenden Datenschutzgesetze dürfen unter dieser Benutzerkennung nicht ohne Sondergenehmigung seitens des RRZE und des Datenschutzbeauftragten verarbeitet werden!
Dem Antragsteller ist bekannt, dass er sich durch eine missbräuchliche Benutzung der Informationsverarbeitungssysteme strafbar machen kann und dass beim Vorliegen eines Missbrauchs grundsätzlich Strafantrag gestellt wird. Des weiteren bemüht sich der Antragsteller, die HPC-Systeme effizient zu nutzen und gängige HPC-Praktiken zu beachten.

Benutzerrichtlinien:
`https://www.rrze.fau.de/infocenter/rahmenbedingungen/richtlinien/benutzungsrichtlinien/`

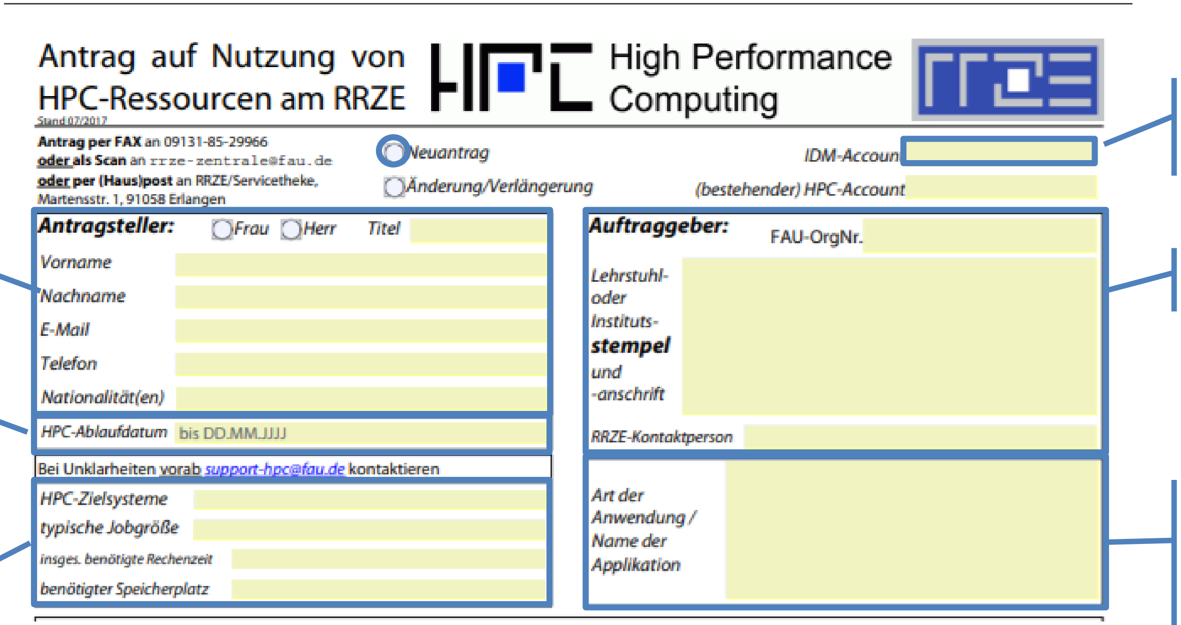
Antragsteller und Auftraggeber erklärt hiermit, von den Benutzungsrichtlinien sowie den ergänzenden Hinweisen auf der Rückseite dieses Antrags Kenntnis genommen zu haben.

Ort, Datum

Unterschrift Antragsteller
Unterschrift Auftraggeber/ Kontaktperson
IdM-Kennung Auftraggeber/Kontaktpers.

RRZE-interne Bemerkungen

HPC account application form (I)



Antrag auf Nutzung von HPC-Ressourcen am RRZE

High Performance Computing

Stand 07/2017

Antrag per FAX an 09131-85-29966
oder als Scan an `rrze-zentrale@fau.de`
oder per (Haus)post an RRZE/Servicetheke,
Martensstr. 1, 91058 Erlangen

○ *Neuantrag*

○ *Änderung/Verlängerung*

IDM-Accoun

(bestehender) HPC-Account

Antragsteller: ○ *Frau* ○ *Herr* *Titel*

Vorname

Nachname

E-Mail

Telefon

Nationalität(en)

HPC-Ablaufdatum bis DD.MM.JJJJ

Bei Unklarheiten vorab *support-hpc@fau.de* kontaktieren

HPC-Zielsysteme

typische Jobgröße

insges. benötigte Rechenzeit

benötigter Speicherplatz

Auftraggeber: FAU-OrgNr.

Lehrstuhl- oder Instituts- ***stempel*** *und -anschrift*

RRZE-Kontaktperson

Art der Anwendung / Name der Applikation

Your IdM account

Your data

Chair data & seal

Account expiration date

Very brief description of what you want to do

Systems, requirements (brief!)

HPC account application form (II)

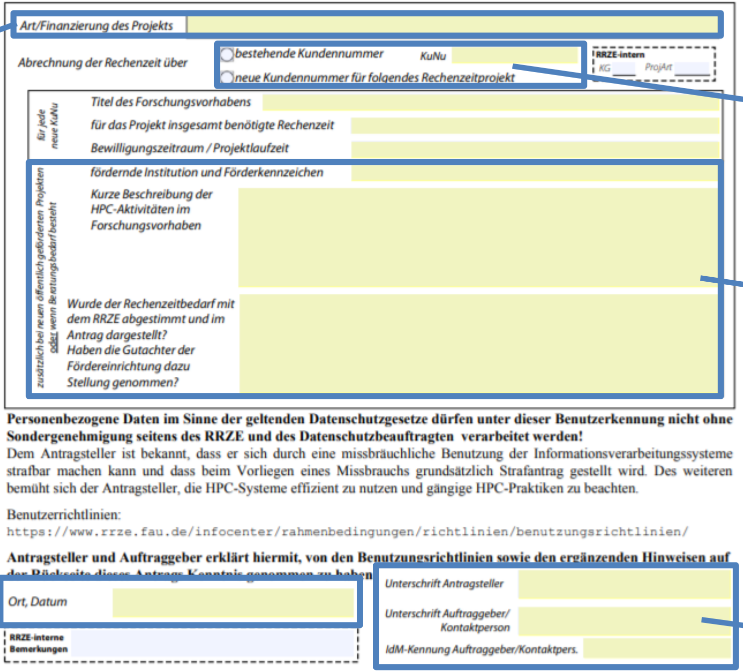
Type of project:

Education (master/bachelor)

Standard (FAU employee)

Research grant (BMBF, DFG, EU)

Industry



Art/Finanzierung des Projekts

Abrechnung der Rechenzeit über
○ bestehende Kundennummer KuNu
○ neue Kundennummer für folgendes Rechenzeitprojekt

RRZE-intern
KG
ProjArt

für jede neue KuNu
Titel des Forschungsvorhabens
für das Projekt insgesamt benötigte Rechenzeit
Bewilligungszeitraum / Projektlaufzeit

zusätzlich bei neuen öffentlich geförderten Projekten oder wenn Beratungsbedarf besteht
fördernde Institution und Förderkennzeichen
Kurze Beschreibung der HPC-Aktivitäten im Forschungsvorhaben
Wurde der Rechenzeitbedarf mit dem RRZE abgestimmt und im Antrag dargestellt? Haben die Gutachter der Fördereinrichtung dazu Stellung genommen?

Personenbezogene Daten im Sinne der geltenden Datenschutzgesetze dürfen unter dieser Benutzerkennung nicht ohne Sondergenehmigung seitens des RRZE und des Datenschutzbeauftragten verarbeitet werden!
Dem Antragsteller ist bekannt, dass er sich durch eine missbräuchliche Benutzung der Informationsverarbeitungssysteme strafbar machen kann und dass beim Vorliegen eines Missbrauchs grundsätzlich Strafantrag gestellt wird. Des weiteren bemüht sich der Antragsteller, die HPC-Systeme effizient zu nutzen und gängige HPC-Praktiken zu beachten.

Benutzerrichtlinien:
`https://www.rrze.fau.de/infocenter/rahmenbedingungen/richtlinien/benutzungsrichtlinien/`

Antragsteller und Auftraggeber erklärt hiermit, von den Benutzungsrichtlinien sowie den ergänzenden Hinweisen auf der Rückseite dieses Antrags Kenntnis genommen zu haben

Ort, Datum

RRZE-interne Bemerkungen

Unterschrift Antragsteller

Unterschrift Auftraggeber/ Kontaktperson

IdM-Kennung Auftraggeber/Kontaktpers.

RRZE customer ID (ask contact person)

More detailed info on third-party funded projects

Date & place

Signatures: you and your boss or contact person

Is there a fee for compute cycles?

- CPU cycles are free for FAU-funded research and education
  - No special permissions, priorities, quotas,…
- DFG/BMBF projects etc.
  - Consult with HPC@RRZE before submitting the DFG proposal
- Industry
  - Set up contract with RRZE
  - Case by case basis
  - There is an official price list: https://www.rrze.fau.de/infocenter/preise-kosten/#hpc

Cluster access

- Primary point of contact: cluster frontends
  - `woody.rrze.uni-erlangen.de` (also for TinyX)
  - `emmy.rrze.uni-erlangen.de`
  - `meggie.rrze.uni-erlangen.de`
  - Only available from within FAU (private IP addresses)

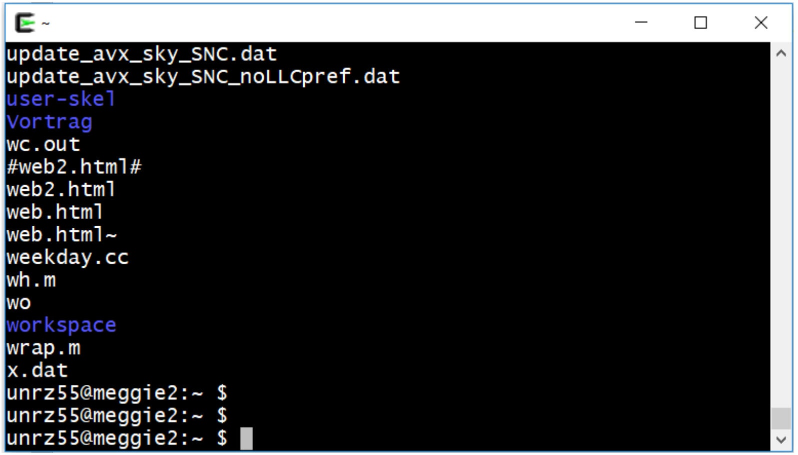
- Access from outside FAU: dialog server
  - `cshpc.rrze.uni-erlangen.de`
  - The only machine with a public IP address

Secure Shell

- By default: text mode only

```
$ ssh ihpc02h@emmy.rrze.uni-erlangen.de
```

- Basic knowledge of file handling, scripting, editing, etc. under Linux is required
- X11 forwarding with option **`-X`** or **`-Y`**
  - Requires local X server



```
update_avx_sky_SNC.dat
update_avx_sky_SNC_noLLCpref.dat
user-skel
Vortrag
wc.out
#web2.html#
web2.html
web.html
web.html~
weekday.cc
wh.m
wo
workspace
wrap.m
x.dat
unrz55@meggie2:~ $
unrz55@meggie2:~ $
unrz55@meggie2:~ $
```

Secure Shell client programs

- Linux: OpenSSH available in any distribution
- Mac: ditto
- Windows
  - Putty (`https://putty.org/`)
  - OpenSSH via Command/PowerShell
  - Linux Subsystem for Windows
  - WinSCP (data transfer only) (`https://winscp.net`)
  - MobaXterm (`https://mobaxterm.mobatek.net/`)
    - includes an embedded X server

Working with data

https://hpc.fau.de/systems-services/systems-documentation-instructions/hpc-storage/

File systems

- File system == directory structure that can store files
- Several file systems can be “mounted” at a compute node
  - Similar to drive letters in Windows (C:, D:, …)
  - Mount points can be anywhere in the root file system

- Available file systems differ in size, redundancy and how they should be used

RRZE file systems overview

Mount point	Access	Purpose	Technology	Backup	Snap-shots	Data lifetime	Quota
/home/hpc	$HOME	Source, input, important results	NFS on central servers, small	YES	YES	Account lifetime	50 GB
/home/vault	$HPCVAULT	Mid-/long-term storage	Central servers	YES	YES	Account lifetime	500 GB
/home/woody	$WORK	Short-/mid-term storage, General-purpose	Central NFS server	(NO)	NO	Account lifetime	330 GB
/*lxfs	$FASTTMP (only within cluster)	High performance parallel I/O	Lustre parallel FS via InfiniBand	NO	NO	High watermark	Only inodes
/???	$TMPDIR	Node-local job-specific dir	HDD/SDD/ ramdisk	NO	NO	Job runtime	NO

File system quotas

- File system may impose quotas on
  - Stored data volume
  - Number of files and directories (inodes, actually)
- Quotas may be set per user or per group (or both)
- Hard quota
  - Absolute upper limit, cannot be exceeded
- Soft quota
  - May be exceeded temporarily (e.g., for 7 days – grace period)
  - Turns into hard quota at end of grace period

Displaying the quota limits

```
$ quota –s           # generic command
Disk quotas for user unrz55 (uid 12050):
     Filesystem  blocks   quota   limit   grace   files   quota   limit   grace
10.28.20.201:/hpcdatacloud/hpchome/shared
                 5544M  51200M    100G           72041    500k   1000k
wnfs1.rrze.uni-erlangen.de:/srv/home
                 112G   318G      477G            199k       0       0

$ shownicerquota.pl  # only on RRZE systems
   Path              Used     SoftQ    HardQ    Gracetime  Filec    FileQ    FiHaQ    FileGrace

   /home/hpc            5.7G    52.5G    104.9G       N/A      72K     500K   1,000K        N/A
   /home/woody          112G   333.0G    499.5G       N/A     188K                          N/A
```

Data transfer

- Most RRZE file systems are mounted at all HPC systems
  - Exception: parallel FS and node-local storage
- No NFS mounting from or to systems outside of RRZE

- → scp / rsync is the preferred file transfer tool from and to the outside

Recurse into subdirectories

Preserve time stamps and access modes

```
$ scp -r -p code unrz55@emmy.rrze.fau.de:/home/woody/unrz/unrz55
$ scp unrz55@emmy.rrze.fau.de:results/output.dat .
```

- Windows: https://winscp.net/

Software

https://hpc.fau.de/systems-services/systems-documentation-instructions/environment/

Pre-installed software packages

Linux standard distro packages

- Cluster front-ends: “Full” installation available, easy to add additional packages
- Node installation: usually stripped down, not easy to add new software

The modules system

- Software provided locally by RRZE
  - Compilers, libraries, commercial and open software
  - Installed on central server and available on all cluster nodes

- A package must be made available in the user’s environment to become usable
  - Command: `module`
  - All module commands affect the current shell only!

The module command

Show available modules: module avail

```
$ module avail
--------------------- /apps/modules/data/applications -----------------------------------------------
amber-gpu/14p13-at15p06-gnu-intelmpi5.1-cuda7.5 gromacs/4.6.6-mkl-IVB
amber-gpu/16p04-at16p10-gnu-intelmpi5.1-cuda7.5 gromacs/5.0.4-mkl-IVB
amber/12p21-at12p38-intel16.0-intelmpi5.1       gromacs/5.1.1-mkl-IVB_d
---------------------- /apps/modules/data/development -----------------------------------------------
cuda/7.5                          intel64/16.0up04                  intelmpi/5.1up03-intel
cuda/8.0                          intel64/17.0up05(default)         llvm-clang/3.8.1
cuda/9.0                          intel64/18.0up02                  opencl/intel-cpuonly-5.2.0.10002
cuda/9.1                          intel64/18.0up03                  openmpi/1.08.8-gcc
$
```

The module command

Load a module: **module load <modulename>**

```
$ module load intel64
$ icc –V
Intel(R) C Intel(R) 64 Compiler for applications running on Intel(R) 64, Version 17.0.5.239 Build
20170817
Copyright (C) 1985-2017 Intel Corporation.  All rights reserved.
```

Display loaded modules: **module list**

```
$ module list
Currently Loaded Modulefiles:
  1) torque/current      2) intelmpi/2017up04-intel      3) mkl/2017up05      4) intel64/17.0up05
```

Module command summary

Command	What it does
module avail	List available modules
module whatis	Shows over-verbose listing of all modules
module list	Shows which modules are currently loaded
module load <pkg>	Loads module pkg, i.e., adjusts environment
module load <pkg>/<version>	Loads specific version of pkg instead of default
module unload <pkg>	Undoes what the load command did
module help <pkg>	Shows a detailed description of pkg
module show <pkg>	Shows what environment variables pkg modifies and how

https://hpc.fau.de/systems-services/systems-documentation-instructions/environment/#modules

Running jobs

https://hpc.fau.de/systems-services/systems-documentation-instructions/batch-processing/

Interactive runs on the front-ends

- The cluster frontends are for interactive work
  - Editing, compiling, preparing input,…
  - Amount of compute time per binary is limited by system limits
    - E.g., after 1 hour of CPU time your process will be killed
  - MPI jobs are not allowed on front ends at RRZE
- Front-ends are shared among all users, so be considerate!

```
iww042@meggie1$ emacs Makefile
iww042@meggie1$ make all
iww042@meggie1$ ./scripts/preprocess.py < inputfile
iww042@meggie1$ ./bin/a.out arg1 arg2 arg3
```

Batch jobs

- All big clusters have resource manager software → “Batch system”
  - Users can request resources for their jobs
    - Number of nodes (optionally: type of nodes, memory, …)
    - Job runtime
    - What to run (normally a shell script)
  - Job will run when resources become available
  - What you do with your node allocation is entirely up to you

- Popular batch systems: PBS Pro, Torque, SLURM, LSF, GridEngine
- Some setups (e.g., at RRZE) allow interactive batch jobs
- Most queues at RRZE have a 24 hour wall time limit

Example: Simple Torque batch script

- Most simple batch script (job1.sh):

```
#!/bin/bash -l
~/bin/a.out arg1 arg2 arg3
```

- Submission:

```
iww042@emmy1$ qsub –l nodes=1:ppn=40,walltime=01:00:00 job1.sh

1051341.eadm
```

Example: Complex Torque batch script

```
#!/bin/bash -l
#PBS -l nodes=4:ppn=40,walltime=06:00:00
#PBS -N Sparsejob_33

# jobs always start in $HOME: change to a temporary job dir on $WOODYHOME
mkdir ${WORK}/$PBS_JOBID
cd ${WORK}/$PBS_JOBID
# copy input file from location where job was submitted, and run
cp ${PBS_O_WORKDIR}/inputfile .
/apps/rrze/bin/mpirun –npernode 20 ${HOME}/bin/a.out -i inputfile -o outputfile
# save output
mkdir -p ${WORK}/output/$PBS_JOBID
cp outputfile ${WORK}/output/$PBS_JOBID
cd
# get rid of the temporary job dir
rm -rf ${WORK}/$PBS_JOBID
```

Job submission options:
Nodes, cores per node, time, name,…

Job option sentinel

$PBS_* variables contain job-relevant data

Actual run of your binary

Example: Managing a Torque job

- Job ID can be used to check and control the job later

```
iww042@emmy1$ qsub job2.sh
1051342.eadm
```

```
iww042@emmy1$ qstat –a
eadm:
                                                                                  Req'd    Req'd       Elap
Job ID                  Username    Queue    Jobname          SessID  NDS   TSK   Memory   Time    S   Time
----------------------- ----------- -------- ---------------- ------ ----- ------ ------ --------- - ---------
1051342.eadm            iww042      devel    test.sh             --       1     40    --    00:10:00 R  00:00:02

iww042@emmy1$ qalter –l walltime=02:00:00 1051342
iww042@emmy1$ qdel 1051342
```

- stdout/stderr will be in **`<JobName>.[o|e]<JobID>`**

Torque user commands (non-exhaustive)

Command	Purpose	Options
qsub [<options>] [-I|<job_script>]	Submit batch job (-I = interactive)	-l <resource_spec> -N <JobName> -o <stdout_filename> -e <stderr_filename> -q <queue_name> -M your@email.de –m abe -X X11 fowarding
qstat [<options>] [<JobID>|<queue>]	Check job status	-a friendly formatting -f verbose job info -r only running jobs -n show nodes of each job
qdel <JobID>	Delete batch job	–

Interactive batch job with Torque

```
iww042@emmy1$ qsub –l nodes=2:ppn=40,walltime=01:00:00 -I
qsub: waiting for job 1051378.eadm to start
qsub: job 1051378.eadm ready

Starting prologue... Mon Jan 28 15:55:44 CET 2019
Master node: e0104
Kill all process from other users
Adjust oom killer config
Clearing buffers and caches on the nodes.
Power management available, enabling ondemand governor
End of prologue: Mon Jan 28 15:55:51 CET 2019
iww042@e0104$
```

Some resources reserved for small jobs during working hours

Mostly harmless ☺

Type stuff here

Some Dos and don’ts

Good practices

- Be considerate. Clusters are valuable shared resources that have been paid by the taxpayer.
- Use the appropriate amount of parallelism
  - Most workloads are not highly scalable
  - Best to run scaling experiments to figure out “sweet spot”
- Check your jobs regularly
  - Are the results OK?
  - Does the job actually use the allocated nodes in the intended way? Does it run with the expected performance?
  - Memory consumption? Disk quota exceeded?
  - Job Monitoring: https://www.hpc.rrze.fau.de/HPC-Status/job-info.php

Good practices

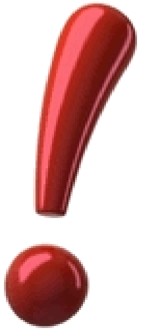
- Use the appropriate file system(s)
  - #1 mistake: Overload metadata servers by doing tiny-size, high-frequency I/O to parallel FS
  - Delete obsolete data

- Do not re-use other people’s job scripts if you don’t understand them completely
  - Things to look out for: file systems, number of nodes, cores per node, modules

- Look at tips and tricks for various applications (e.g. example batch scripts): https://hpc.fau.de/systems-services/systems-documentation-instructions/special-applications-and-tips-tricks/

Good practices

- Learn a scripting language to automate daunting, repetitive tasks
  - Bash, Python, Perl,…
- Talk to co-workers who are more experienced cluster users; let them educate you
- When reporting a problem to RRZE:
  - Use the official contact hpc-support@fau.de – this will immediately open a helpdesk ticket
  - Provide as much detail as possible so we know where to look
    - “My jobs always crash” will not do
    - Cluster, JobID, file system, time of event, …



THANK YOU.

HPC@RRZE

https://hpc.fau.de