Course Information

• The aim of this course is to provide an introduction to the High Performance Computing (HPC) infrastructure of the Leibniz Supercomputing Centre (LRZ)

• You will probably benefit the most, if you’re not yet familiar with the LRZ HPC infrastructure, but plan to work on these systems in the future
  -> by the end of this workshop, you should have the basic skills to successfully interact remotely with LRZ HPC systems

• Consider the following – especially during hands-on sessions:
  -> you may want to partner up with the person sitting next to you
  -> it may be beneficial to sit back and watch the slides/demos
  -> the slides will be made available after the workshop
  -> generally: please ask, if you have any questions
Moore’s Law

The number of transistors on integrated circuits is doubling about every two years.

CC-BY-SA Max Roser; Data source: https://en.wikipedia.org/wiki/Transistor_count
Introduction to the LRZ HPC Infrastructure | 08/10/2019 | J. Albert-von der Gönna
Mid 2000s: “heat death”

No more faster processors, only more of them.

But: 2x3 GHz ≠ 6 GHz
TOP500 List

Performance Development

From #1 to #500: 6-8 years

From #500 to Notebook: 8-10 years
Aggregated LRZ Systems

Evolution of Peak Performance and Memory

Introduction to the LRZ HPC Infrastructure | 08/10/2019 | J. Albert-von der Gönna
What is a Supercomputer… (not)?

- It has overclocked high-speed processors? **Nope**
- The CPU runs faster than a desktop PC? **Nope**
- It has a large internal memory (RAM)? **Nope (with some exceptions)**
- It runs Microsoft Windows? **Nope**
- It will run my Excel spreadsheet? **Nope**
- It will run my old tried and tested executable? **Probably not**
- It will run my software without changes? **Probably not**
- It will run my program with millions of threads? **Nope**
- It can be used interactively? **Mostly no**
So… what is a Supercomputer?

- It consists of many off-the-shelf (server) CPUs with vector instructions (e.g. AVX2, AVX512) in login and compute nodes (as well as service/management nodes)
- All these nodes are connected by a high-speed internal network (interconnect, e.g. InfiniBand, OmniPath)
- They are diskless but have access to a parallel file system (e.g. Lustre, GPFS)
- The compute nodes can generally not be accessed directly, but programs have to be submitted to a batch scheduler application (e.g. LoadLeveler, Slurm) from the login nodes (which are usually accessible by SSH)
- Communication and parallelization is typically relying on the message passing interface standard (MPI) between nodes and the Open Multi-Processing API (OpenMP) on individual nodes
- The operating system is Unix-like, i.e. GNU/Linux
HPC Cluster Systems

Pruned Tree

Switch

Fat Tree

Node

Accelerator: GPU, FPGA

Socket

Core

Island
Levels of Parallelism

- Node Level (e.g. SuperMUC-NG has 6480 nodes)
- Accelerator Level (e.g. Nvidia DGX-1 has 8 GPUs)
- Socket Level (e.g. Linux Cluster Teramem has 4 Sockets/CPUs with 25 cores each)
- Core Level (e.g. Linux Cluster CoolMUC-3 nodes have 64 cores)
- Vector Level (e.g. AVX-512 has 32 512-bit vector registers)

- SuperMUC-NG Peak Performance: \( 26,87 \text{ PFlop/s} = 6480 \text{ Nodes} \times 2 \text{ Sockets} \times 24 \text{ Cores} \times 32 \text{ Vectors} \times 2,7 \text{ GHz} \)
SuperMUC-NG: High Level System Architecture

Pruning ratio is **3.75 : 1**

- **Storage Island (NSR0)**: 25 racks (258 downlinks)
- **Service + Fat Node Island**: 2 racks (144 fat nodes), 54 Login/Service/Mgmt
- **Island 1**: 11 racks (792 nodes)
- **Island 8**: 11 racks (792 nodes)

**OPA1 DCS**
- 1152 ports, 384 used
- 1152 ports, 324 used
- 1152 ports, 1002 used

**OPA1 Core switches**
- 42 (48 ports each)
## SuperMUC-NG: Hardware Overview

<table>
<thead>
<tr>
<th>Name</th>
<th>CPU</th>
<th>Cores/Node</th>
<th>RAM/Node (GB)</th>
<th>Nodes</th>
<th>Cores</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuperMUC-NG Thin Nodes</td>
<td>Intel Xeon (&quot;Skylake&quot;)</td>
<td>48</td>
<td>96</td>
<td>6336</td>
<td>304128</td>
</tr>
<tr>
<td>SuperMUC-NG Fat Nodes</td>
<td>Intel Xeon (&quot;Skylake&quot;)</td>
<td>48</td>
<td>768</td>
<td>144</td>
<td>8912</td>
</tr>
</tbody>
</table>
HPC Systems for Bavarian Universities

CoolMUC-2 Teramem CoolMUC-3 IvyMUC

DGX-1 (P) DGX-1 (V) 4xP100

Compute Cloud (OpenStack)

Tape Archive and Backup

DSS (Data Science Storage)

lxlogin8.lrz.de
 lxlogin[5-7].lrz.de lxlogin10.lrz.de https://datalab.srv.lrz.de

https://www.rstudio.lrz.de

https://cc.lrz.de
## Linux Cluster: Hardware Overview

<table>
<thead>
<tr>
<th>Name</th>
<th>CPU</th>
<th>Cores/Node</th>
<th>RAM/Node (GB)</th>
<th>Nodes (total)</th>
<th>Cores (total)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoolMUC-2</td>
<td>Intel Xeon E5-2690 v3 (&quot;Haswell&quot;)</td>
<td>28</td>
<td>64</td>
<td>384</td>
<td>10752</td>
</tr>
<tr>
<td>CoolMUC-3</td>
<td>Intel Xeon Phi (&quot;Knights Landing&quot;)</td>
<td>64</td>
<td>96</td>
<td>148</td>
<td>9472</td>
</tr>
<tr>
<td>IvyMUC</td>
<td>Intel Xeon E5-2650 (&quot;Ivy Bridge&quot;)</td>
<td>16</td>
<td>64</td>
<td>31</td>
<td>496</td>
</tr>
<tr>
<td>Teramem</td>
<td>Intel Xeon E7-8890 v4 (&quot;Broadwell&quot;)</td>
<td>96</td>
<td>6144</td>
<td>1</td>
<td>96</td>
</tr>
</tbody>
</table>
Linux Cluster: Access

• In order to use LRZ services provided to Bavarian universities, an LRZ account (belonging to an LRZ project) with appropriate permissions is needed. Student/staff accounts from LMU and TUM are restricted to certain services (e.g. E-Mail, Cloud Storage, LRZ Sync+Share) and can not be used to obtain Linux Cluster access.

• Department/institute heads and/or professors/PIs can request new LRZ projects and appoint a master user (or more) for the project. The master user(s) can manage accounts and permissions within these LRZ projects.

• If such an LRZ project already exists for your department (or institute, or research group), contact the master user and ask for an account with Linux Cluster permissions. If not, see https://doku.lrz.de/x/TQAOAQ
SuperMUC-NG: Access

There are three (four) ways to apply for using SuperMUC-NG:

- GCS test project: rolling call, fast review (short abstract), < 100,000 CPUh
- GCS regular project: rolling call, technical & scientific review, < 35m CPUh
- GCS large scale project: twice per year, technical & scientific review, > 35m CPUh
- (PRACE projects for academic users from other European countries)

See https://doku.lrz.de/x/XAAAbAQ
Data Storage

• The LRZ HPC Infrastructure is backed by the Data Science Storage (DSS)
  • Long-term storage solution for potentially vast amounts of data
  • Directly connected to the LRZ computing ecosystem
  • Flexible data sharing among LRZ users
  • Web interface for world-wide access and transfer
  • Data sharing with external users (invite per e-mail, access per web interface)
• Disk space and access is managed (as containers) by data curators. This can be LRZ personnel (e.g. Linux Cluster $HOME directories) or PIs/master users (e.g. data projects).
Data Storage

• $HOME (home directory)
  • 100GB per project, i.e. this quota applies to all accounts of the project combined (!)
  • Automatic backup and snapshots
    (there is a truly hidden “$HOME/.snapshot” subdirectory)
  • All your important files/anything you invested a lot of work into should be here
  • This will be deprecated by the end of the year!
    On November 11th (schedule may change, watch out for announcements) new DSS-based storage will become the default
    (and the above will become $HOME_LEGACY).
Data Storage

- $PROJECT (DSS-based project file system)
  - 10TB per project, can be extended (by at least 20TB) as paid option
  - Configuration (e.g. exports, backup) to be managed by data curator (!)
  - Use this for e.g. large raw data (and consider backup options)

- $PROJECT_LEGACY (deprecated, read-only project file system)
  - Initially 1TB per project, can be extended to 5TB (or more) if requested
  - No (automatic) backup by LRZ (!)
  - Use this for e.g. large raw data with a backup somewhere else
    (either at your institution or the LRZ tape archive)
Data Storage

• $SCRATCH (scratch file system, “temporary file system”)
  • Several thousand TBs, i.e. PBs
  • No backup (!) and sliding window file deletion, i.e. old files will eventually be deleted (!!!)
    – a data retention time of approx. 30 days may be assumed, but is not guaranteed
  • This is the place for e.g. very large, temporary files or intermediate results, directly
    feeding into additional analyses
  • Do not save any important data exclusively on this file system!
    Seriously, don’t do it!

• Tape archive: visualize industrial robots juggling tape drives.
  If this is what you need, take a look at the documentation and/or talk to us.
Collaboration Tools

- Videoconferencing
  https://meet.lrz.de
- Git-repository management, issue tracker, wiki…
  https://gitlab.lrz.de
• Testing it here might break the Wi-Fi…
  explore it at home with your friends!
• Use a modern browser
  (Safari will very likely not work)
Video Conferencing

https://meet.lrz.de
Video Conferencing

https://meet.lrz.de
- Ever heard of Microsoft’s proprietary platform called GitHub?
  Yeah, it’s very much like that.
- Every LRZ project member can invite up to 20 external users to the service (GitInvited), this is great for collaborations!
Git Repository Management

https://gitlab.lrz.de

Introduction to the LRZ HPC Infrastructure | 08/10/2019 | J. Albert-von der Gönna
Git Repository Management

https://gitlab.lrz.de
• Web-based RStudio frontend
• Cluster of multiple nodes, with
  • 8 cores and
  • 80 GB RAM each
    (upgrade pending, watch out for announcements)
• Connected to the same (DSS-)file systems like the Linux Cluster
R version 3.4.4 (2017-11-30) -- "Kite-Eating Tree"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
Natural language support but running in an English locale
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'training()' on how to cite R or R packages in publications.
Type 'demo()' on how to use R for some advanced features.
Type 'help', 'help.search', 'help.start', 'pkgdoc()' for help.

Let’s get started:
Log in to the RStudio Server and get familiar with the interface
Create a new folder in your home directory
(bonus points for using the built-in terminal)
• Integrated Terminal: Provides access to the system shell from within RStudio
• At the moment, you need to disable WebSocket support: Tools -> Global Options… -> Terminal
Let’s get started:
  Log in to the RStudio Server and get familiar with the interface
• Create a new folder in your home directory
  (bonus points for using the built-in terminal)
• Create a new text file, write down something nice and save it to the newly created folder
• Make sure you can locate the folder/file in the file system
  (using the Files pane and/or the terminal)
• Extra credit: what is the full path to your files on the file system?
Linux Cluster

- Connect to the CoolMUC-2 segment of the Linux Cluster

- From a terminal application:
  
  $ ssh <user>@lxlogin5.lrz.de
For CoolMUC-2 you can use the login nodes
lxlogin5.lrz.de or
lxlogin6.lrz.de or
lxlogin7.lrz.de

(see documentation for other cluster segments/systems)
Linux Cluster – CoolMUC-2

What happened? What to do?
Interrupt by pressing Ctrl+C
• Using a local terminal, add your SSH public key (not the private one!) to the authorized keys on the Linux Cluster!
• On Linux, this is the content of ~/.ssh/id_rsa.pub on your local machine…
• … which should go into ~/.ssh/authorized_keys on the login node
• Use the command `ssh-copy-id <user>@lxlogin5.lrz.de` on Linux, you will have to do it manually on macOS and Windows (but only once)
Linux Cluster – CoolMUC-2

```
di36pez@ubuntu1804:~$ ssh di36pez@lxlogin5.lrz.de
The authenticity of host 'lxlogin5.lrz.de (129.187.20.105)' can't be established.

ECDSA key fingerprint is SHA256:YmtVvUOlDqTzoZ4piDC4encMuJe8WIjJuA4NqmAeXgX.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'lxlogin5.lrz.de,129.187.20.105' (ECDSA) to the list of known hosts.
Password:
```
di36pez@ubuntu1804:~$ ssh-copy-id di36pez@lxlogin5.lrz.de
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
Password:
```
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'di36pez@lxlogin5.lrz.de'"
and check to make sure that only the key(s) you wanted were added.
```
```
• Connect to the cluster again, it should now work without password! Crikey!
• You can use the `logout` or `exit` commands to close the connection
Welcome to the CoolMUC2 Infiniband cluster, one of the Linux cluster systems operated by Leibniz Supercomputing Centre (LRZ).

Please read the introductory documentation on the LRZ web site http://www.lrz.de/services/compute/linux-cluster

In case of problems please contact the LRZ Service Desk via https://servicedesk.lrz.de/
(select the service "Hochleistungsrechnen --> Linux Cluster")

Please do not run any extensive computational programs on the front end node. Instead, please submit SLURM batch scripts for production jobs, and SLURM interactive shells for testing and short-running programs.

Misuse of the interactive resources will lead to violating accounts being blocked from access to the cluster.

Please go to https://doku.lrz.de/display/PUBLIC/High+Performance+Computing for current messages and the system status.

Home directories are using 10980 GB out of 12929 GB, 1949 GB are available
Project directory is using 1837 GB out of 2100 GB, 264 GB are available
• Get your bearings… where did you end up on the file system? (Hint: $ pwd)
• Can you locate the folder and file created earlier (with RStudio Server)?
Linux Cluster – CoolMUC-2

• Take a look at the contents of your file
  
  $ cat my_file

• Look at and explore $HOME (and $PROJECT) (these are environment variables, remember?)
  
  $ echo $HOME
  $ echo $PROJECT
Environment Modules

- Modules allow for the dynamic modification of environment variables, e.g. they provide a flexible way to access various applications and libraries.
- List the currently active modules (loaded by default):
  
  ```bash
  $ module list
  ```
Environment Modules

- Get more info about e.g. the lrz module:
  
  $ module show lrz
Environment Modules

```
dl36pez@mpp2-login5:~$ module show lrz

/lrz/sys/share/modules/files/environment/lrz/default:

module-whatis  Environment: Default setup for all LRZ users
setenv        LRZ_ARCH   x86_64_Intel
setenv        LRZ_SYSTEM  Cluster
append-path   PATH /lrz/sys/tools/slurm_utils/bin
setenv        SALLOC_PARTITION  mpp2_intcr
setenv        LRZ_SYSTEM_SEGMENT CMUC2
setenv        LRZ_SUB_ARCH    Haswell_EP
setenv        LRZ_INSTRSET    x86_avx2
setenv        LRZ_OS         SUSE Linux Enterprise Server 12 (x86_64)
setenv        LRZ_OS_VER     12
setenv        LRZ_OS_SUBVER  3
setenv        LRZ_NOCHECK   yes
setenv        INTEL_LICENSE_FILE /lrz/sys/intel/licenses

dl36pez@mpp2-login5:~$ 
```
Environment Modules

• Suppose you need to use MATLAB
• It is not generally available (try `$ which matlab`)
• … or is it?
  Search for available modules:
  `$ module available matlab` or
  `$ module av matlab`
Environment Modules
Environment Modules

• Look at all these options…!
• We’re currently in a transition phase: going forward (most, if not) all modules will be maintained using the Spack package manager, i.e. prioritize modules in the “/lrz/sys/spack/…” path!
• Load any module you like, e.g. the latest MATLAB version:
  $ module load matlab/R2018b-intel
Environment Modules

```
[...]
dl36pez@mpp2-login5:~$ module load matlab/R2018b-intel

WARNING: Please note that the dynamic loader is overloaded by this MATLAB environment module!

Please note further that the setting of the KMP_AFFINITY environment variable is also modified by MATLAB environment module! This may have negative impact on the performance and functionality of other OpenMP based programs.

Use a different shell to start programs other than MATLAB, otherwise those programs may not function properly.

dl36pez@mpp2-login5:~$ which matlab
/lrz/mnt/sys.x86_sles12/spack/18.2/opt/x86_avx2/matlab/R2018b-intel-e343tgbl/bln/matlab
 dl36pez@mpp2-login5:~$ [...]
```
Environment Modules

• Modules can/should be unloaded when you don’t need them anymore (e.g. before trying another version):
  
  $ module unload <module/version>

• Loading modules is not persistent across sessions, i.e. once you log out and back in again, only the default modules will be loaded!
Slurm Workload Manager

• Slurm is a job scheduler:
  • Allocates access to resources (time, memory, nodes/cores)
  • Provides framework for starting, executing, and monitoring work
  • Manages queue of pending jobs (enforcing “fair share” policy)
• Use the *sinfo* command to get information about the available clusters
  $ sinfo --clusters=all or, shortened:
  $ sinfo -M all
**Slurm Workload Manager**

- Look for the cluster segments
  - inter (allows for interactive usage)
  - mpp2 (the main CoolMUC-2 cluster)
  - serial (shared nodes for serial jobs)
- What is their current status?
- Get information about a specific cluster segment, e.g.
  
  $ \texttt{sinfo -M inter} \text{ or } \texttt{sinfo -M mpp2}$
Interactive Allocation

- The inter cluster can be used for interactive resource allocation:
  
  ```
  $ salloc -p mpp2_inter -n 1
  ```

- Once the resources are made available, you can e.g. start a terminal process on the allocated node:
  
  ```
  $ srun --pty bash -i
  ```
Interactive Allocation

- Notice the change of the hostname, you’re now logged in on a compute node!
- For production jobs, you want to prepare and submit batch scripts – they tell Slurm about the resources you need and the scripts/programs you want to run.
Job Processing

#!/bin/bash
#SBATCH --clusters=mpp2
#SBATCH --nodes=1
module load slurm_setup
./<executable>

• A very minimal example of a job script (not necessarily recommended, but working in some cases), requesting
  • a single, exclusive node (with 28 CPUs)
  • of the the CoolMUC-2 cluster segment mpp2
• Submit this job script to the queue:
  $ sbatch <script.sh>
#!/bin/bash
#SBATCH -o /home/hpc/.../.../myjob.%j.%N.out
#SBATCH -D /home/hpc/.../.../workdir
#SBATCH -J jobname
#SBATCH --get-user-env
#SBATCH --clusters=mpp2
#SBATCH --nodes=1
#SBATCH --mail-type=end
#SBATCH --mail-user=xyz@xyz.de
#SBATCH --export=NONE
#SBATCH --time=08:00:00
module load slurm_setup

cd workdir
./<executable>

- A more practical example...
  - defining custom output file(s)
  - setting a working directory
  - assigning a job name
  - configuring mail notifications
  - managing the environment
  - limiting walltime explicitly
- See documentation for more options
Job Processing – Give it a try…

#!/bin/bash
#SBATCH --clusters=serial

module load slurm_setup

hostname

• Create a new folder in your home directory (e.g. “tmp”) and change into it.
• Create this very, very minimal example of a job script and save it as “script.sh”. What will it do?
• Submit this job script to the queue: $ sbatch script.sh
• Keep your eyes open for output. What can you find?
Job Queue and Accounting

- Use the `squeue` command to query information about your jobs in the Slurm scheduling queue, e.g. of the mpp2 cluster:
  
  ```
  $ squeue -M mpp2 -u <user>
  ```

- If you’re interested in the approx. start time of your pending jobs (in the the mpp2 queue):
  
  ```
  $ squeue -M mpp2 -u <user> --start
  ```

- Display accounting data of (finished) jobs by use of the `sacct` command, e.g.
  
  ```
  $ sacct -M mpp2 -u <user>
  ```

- Per default, this is limited to today’s jobs, add the `-S` option to specify a user-defined date:
  
  ```
  $ sacct ... -S <YYYY-MM-DD>
  ```