

Practical Example: High Performance Computing

Ivan Gesteira Costa & Mingbo Cheng & Martin Manolov &
James Nagai & Mina Shaigan

Institute for Computational Genomics

RWTH Compute Cluster

- High-Performance-Computing infrastructure

- Linux-based

- Highly paralellizable



- All RWTH affiliates are granted access, but with limited resource allocation
- More information on the following [link](#).

RWTH RegApp

- Use RWTH RegApp (<https://regapp.itc.rwth-aachen.de/>) to create and manage an HPC-account to use the cluster

RegApp

You have already registered with the following services:

RWTH High Performance Computing

The IT Center hosts one of the fastest supercomputers in Germany. The High Performance Computing group (HPC) supports users from all German universities including institutions from RWTH Aachen University in the efficient use of the central high-performance computing systems.

📄 [Service description](#)

🔑 [Registry info](#)

🔑 [Set service password](#)

🔑 [Set SSH Key](#)

RWTH RegApp 2FA Tokens

- Additionally to establishing a password, a user should establish an MFA Token and/or SSH Key. We will proceed in this tutorial to use MFA Tokens ([Link](#)).
- After RegApp Login, navigate to Index -> My Tokens
- * Use your configured tokens to authenticate and create a new one for the HPC
- If you don't have any tokens configured, follow the instructions to create a TAN List to use to authenticate once (and don't use it again unless its an emergency)

SSH and Cluster Login (subject to change*)

Server-name	OS	Purpose
login18-N.hpc.itc.rwth-aachen.de	Rocky Linux 8 (Nr 4 is CentOS)	Front-End Dialogue System CLAIX2018 $N=\{1,2,3,4\}$
login18-x-N.hpc.itc.rwth-aachen.de	Rocky Linux 8	Remote Desktop sessions $N = \{1,2\}$
login18-g-N.hpc.itc.rwth-aachen.de	Rocky Linux 8	GPU-System $N = \{1,2\}$
lect.hpc.itc.rwth-aachen.de lect2.hpc.itc.rwth-aachen.de	Rocky Linux 8	Dedicated to attend lectures
login23-N.hpc.itc.rwth-aachen.de	Rocky Linux 8	Front-End Dialogue System CLAIX2023 $N=\{1,2,3,4\}$
login23-x-N.hpc.itc.rwth-aachen.de	Rocky Linux 8	CLAIX2023 Remote Desktop sessions $N = \{1,2\}$
login23-g-1.hpc.itc.rwth-aachen.de	Rocky Linux 8	CLAIX2023 GPU-System

SSH and Cluster Login

- Use established **Secure *Shell* Protocol (*ssh*)** connection to log in to the front-end cluster nodes with your username and password
- `$ ssh <username>@<server-address>`
- In the case of the RWTH cluster:
- `$ ssh <TIM>@login23-N.hpc.itc.rwth-aachen.de`

SSH and Cluster Login

- Special nodes for intensive I/O operations

copy18-1.hpc.itc.rwth-aachen.de	2 x 40 Gbit/s
copy18-2.hpc.itc.rwth-aachen.de	2 x 40 Gbit/s

- Use these nodes if you need to transfer large files

Cluster Available File Systems

Name	Path	Backup	Quota (file)	Quota (#files)
\$HOME	/home/<TIM-Kennung>	yes	150 GB	-
\$WORK	/work/<TIM-Kennung>	no	250 GB	-
\$HPCWORK	/hpcwork/<TIM-Kennung>	no	1000 GB	50000
\$TMP	Local Ext4/XFS storage	no	-	-
\$BEEOND	BeeGFS On Demand	no	-	-

Source code
Configuration files

Output files
Working data

IO intensive job,
large files

Cluster - mount drive

- Use *sshfs* to mount remote directory to local machine:
 - `$ sshfs <TIM>@copy23-1.hpc.itc.rwth-aachen.de:<Path> <local Path>`
- Unmount with:
 - `$ sudo umount -l <local Path>`
- Access to cluster only from eduroam or through RWTH VPN *
 - <https://help.itc.rwth-aachen.de/en/service/vbf6fx0gom76/article/6a2cfd0933604cd28eaaa69194ff8d16/>
 - * VPN also requires 2FA now

Cluster - Batch Jobs

- Job handling system - SLURM
- SBATCH [option] command [arguments]
- Slots

Parameter	Function
-c, --cpus-per-task <numcpus>	Number of threads/processes for an OpenMP/ Hybrid script
-n, --ntasks <numtasks> --ntasks-per-node <numtasks>	Number of threads/processes for an MPI job
-N, --nodes <numnodes>	Number of nodes/hosts for the job

- List of slurm commands:
 - <https://help.itc.rwth-aachen.de/en/service/rhr4fjjuttf/article/3d20a87835db4569ad9094d91874e2b4/>

Cluster - Batch Jobs

- Further job parameters

Parameter	Function
-J --job-name=<jobname>	A name for the current job
--mem-per-cpu=<size>	Required RAM per allocated CPU
-o, --output=<filename>	Name for a report file, containing the standard output of the job
-t, --time=d-hh:mm:ss	Time for job execution, after this time the job is killed
-A, --account=<project>	Submit a job for a specific project
--gres=gpu:<type>:2	Requesting two GPUs per node

- Batch documentation - <https://slurm.schedmd.com/sbatch.html>

Cluster - Batch Jobs

- **sbatch** job.sh - sends job, declared in job.sh, to the scheduler
- After defining the slurm parameters, don't forget to write the script you want to execute
- You job might need to load some modules, update environmental variables or activate conda environments.
 - **module** load Python/3.9.6
 - **module** avail # Lists all available modules
 - **export PATH=~/local/bin:\$PATH**
 - **conda activate myenv**

Cluster - Example Scripts

```
#!/bin/bash
```

```
### Job name
```

```
#SBATCH --job-name=MYJOB
```

```
### File for the output
```

```
#SBATCH --output=MYJOB_OUTPUT
```

```
### Time your job needs to execute, e. g. 15 min 30 sec
```

```
#SBATCH --time=00:15:30
```

```
### Memory your job needs per node, e. g. 1 GB
```

```
#SBATCH --mem=1G
```

```
### The last part consists of regular shell commands:
```

```
### Change to working directory
```

```
cd /home/usr/workingdirectory
```

```
### Execute your application
```

```
myapp.exe
```

Cluster - Monitor Jobs and Resources

- Use **squeue** to monitor current jobs in progress:
 - `squeue -u <TIM> #` to only view your jobs
- Use **scancel** to stop current jobs from executing
 - `scancel -u <TIM> #` cancels all your jobs
 - `scancel -n <JOBID> #` cancels job with ID=JOBID
- Use **quota** to see how much space you're occupying
- Use **r_wlm_usage -q** to view your accounting information
- More info:
 - <https://help.itc.rwth-aachen.de/service/rhr4fjjuttf/article/13ace46cfbb84e92a64c1361e0e4c104/>

Cluster - GPU Nodes

- Login to a GPU node by:
 - **ssh <TIM>@login23-g-1.hpc.itc.rwth-aachen.de**
- See GPU usage with **nvidia-smi**

NVIDIA-SMI 550.54.15				Driver Version: 550.54.15				CUDA Version: 12.4			
GPU	Name			Persistence-M	Bus-Id	Disp.A		Volatile	Uncorr. ECC		
Fan	Temp	Perf		Pwr:Usage/Cap		Memory-Usage		GPU-Util	Compute M.		
									MIG M.		
0	NVIDIA	H100		On	00000000:1B:00.0	Off				0	
N/A	35C	P0		68W / 700W	0MiB / 95830MiB			0%	Default		
									Disabled		
Processes:											
GPU	GI	CI		PID	Type	Process name				GPU Memory	
	ID	ID								Usage	
No running processes found											

Cluster - GPU Nodes

- Login to a GPU node by:
 - **ssh <TIM>@login23-g-1.hpc.itc.rwth-aachen.de**
- See GPU usage with **nvidia-smi**

NVIDIA-SMI 550.54.15			Driver Version: 550.54.15			CUDA Version: 12.4		
GPU	Name		Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr. ECC	
Fan	Temp	Perf	Pwr:Usage/Cap		Memory-Usage	GPU-Util	Compute M.	
							MIG M.	
0	NVIDIA H100		On	00000000:1B:00.0	Off			0
N/A	35C	P0	68W / 700W	0MiB / 95830MiB		0%	Default	Disabled

GPU-id
and type

Processes:							
GPU	GI	CI	PID	Type	Process name	GPU Memory	
	ID	ID				Usage	
No running processes found							

Cluster - GPU Nodes

- Login to a GPU node by:
 - **ssh <TIM>@login23-g-1.hpc.itc.rwth-aachen.de**
- See GPU usage with **nvidia-smi**

NVIDIA-SMI 550.54.15				Driver Version: 550.54.15				CUDA Version: 12.4			
GPU	Name	Perf	Persistence-M	Bus-Id	Disp.A	Memory-Usage	Volatile	Uncorr.	ECC		
Fan	Temp		Pwr:Usage/Cap				GPU-Util	Compute	M.		
								MIG	M.		
0	NVIDIA H100		On	00000000:1B:00.0	Off					0	
N/A	35C	P0	68W / 700W	0MiB / 95830MiB			0%	Default	Disabled		
Processes:											
GPU	GI	CI	PID	Type	Process name					GPU Memory	
	ID	ID								Usage	
No running processes found											

GPU-id
and type

GPU memory

Cluster - GPU Nodes

- Login to a GPU node by:
 - **ssh** <TIM>@login23-g-1.hpc.itc.rwth-aachen.de

- See GPU usage with **nvidia-smi**

Compute model: Default - multiple operations can run

NVIDIA-SMI 550.54.15			Driver Version: 550.54.15			CUDA Version: 12.4		
GPU	Name	Perf	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr. ECC	
Fan	Temp		Pwr:Usage/Cap		Memory-Usage	GPU-Util	Compute M.	MIG M.
0	NVIDIA H100		On	00000000:1B:00.0	Off		0	
N/A	35C	P0	68W / 700W	0MiB / 95830MiB		0%	Default	Disabled

Processes:							
GPU	GI	CI	PID	Type	Process name	GPU Memory	
	ID	ID				Usage	
No running processes found							

GPU-id
and type

GPU memory

Cluster - Exercise - GPU with python

- Sing in to the cluster
 - \$ **ssh** <TIM>@login23-g-2.hpc.itc.rwth-aachen.de
- Check if you're successfully in the group
 - \$ groups #should see lect0116 among others
- Load modules and install some libraries locally
 - \$ **module** load Python/3.9.6
 - \$ **module** load cuDNN/8.6.0
 - \$ **module** load CUDA/11.8
 - \$ **export PATH=~/.local/bin:\$PATH**
 - \$ **pip3** install --user tensorflow pandas scikit-learn
 - \$ **pip3** install --user matplotlib tensorrt numba

Cluster - Exercise - GPU with python

- Create a new directory and copy the files needed:
 - \$ **cd** ~/
 - \$ **mkdir** BioInfo
 - \$ **rsync** -rp /home/lect0116/sample BioInfo
 - \$ **cd** BioInfo/sample
- Edit (**vim**) your email address and submit job (**sbatch** run.sh)
- Check status regularly (\$ **squeue** -u <TIM>)

Practical Example: Linux Basics

The Operating System UNIX

- UNIX Development
 - 60s: MULTICS: MULTiplexed Informaion and Computing Service
 - Developed by MIT, Bell Labs and General Electrics
 - Written in Programming Language 1 (PL1) and Assembly
- Thompson and Ritchie later create UNIX, written in C

Linux – Basic Commands (Revisit)

- command parameter1 parameter2 ...
- *man* – **man**ual. Displays manuals for linux packages (try *man man*). *Displays useful information about how to use the package.*
- *pwd* – **p**rint **w**orking **d**irectory. Displays the current working directory

Linux – Basic Commands (Revisit)

- `/s` – **list**. Displays the content of a directory

- `ls -l ~/Pictures`

-rw-r--r--	1	martin	martin	101685	Apr 29	13:23	Selection_001.png
-rw-r--r--	1	martin	martin	39835	Apr 29	14:37	Selection_002.png
-rw-r--r--	1	martin	martin	76985	Apr 29	14:48	Selection_003.png
-rw-r--r--	1	martin	martin	27096	May 2	14:04	Selection_004.png
-rw-r--r--	1	martin	martin	62783	May 3	11:58	Selection_005.png
-rw-r--r--	1	martin	martin	65201	May 6	16:11	Selection_006.png
drwxrwxr-x	2	martin	martin	4096	Apr 24	11:41	Temp

- | Access rights | Links | Owners/
Groups | Size | Last
Modified | Filename |
|---------------|-------|-------------------|------|------------------|----------|
|---------------|-------|-------------------|------|------------------|----------|

Linux – Basic Commands (Revisit)

- `/s` – **l**ist. Displays the content of a directory

```
-rw-r--r--      1 martin martin 101685 Apr 29 13:23 Selection_001.png
```

- Access rights:
 - Type: (**d**)irectory, (**l**)ink, (**-**) a file
 - Rights: (**r**)ead, (**w**)rite, e(**x**)ecute
 - Sequence: *user group anybody*
- Owner/Group:
 - Owner – first column: *martin*
 - Group – second column: *martin*
- Either can be changed with *chmod* and *chown/chgrp* respectively
- Run executable with full path or from directory with “*./file*”

Linux – Basic Commands (Revisit)

- *cd* – **c**hange **d**irectory. Switches to a new directory, supplied as a parameter
- *mkdir* – **m**ake **dir**ectory. Creates a new directory
- *rm* – **r**emove. Removes a specified file or directory (“-r”)
- *cp* – **c**opy. Copy a file or directory (“-r”)
- *scp* – **s**ecure **c**opy. Copy a file to or from a remote source
- *rsync* – **r**emote **s**ync. A fast copying tool. Also works for remote copy

Linux – Basic Commands (Revisit)

- *ln* – **lin**k. Link a file into a new directory
- *echo* – Prints a string to standard output
- *cat* – con**cat**enate. The content of a file is printed to standard output
- *wc* – **w**ord **c**ount. Counts the number of words, rows (“-l”) or characters (“-c”) in a file.

Linux – Basic Commands (Revisit)

- pipe (“|”) - connect commands
- output (“>”) and input (“<”) for a specific program.
Output can also concatenate to existing content, without deleting (“>>”)

Linux – Environment Variables

Variable	Description
PATH	Colon separated list of directories, which will be searched through when entering a name of executable
HOME	The pathname of the home directory.
SHELL	The currently used shell program
USER	The current username

```
$ echo $PATH  
/usr/local/sbin
```

```
$ export PATH=~/.local/bin:$PATH  
$ echo $PATH  
/home/martin/.local/bin:/usr/local/sbin
```

Linux – Vim

- A further development of Vi
- Open-Source editor for use inside the Terminal
- Open file with “\$ vim filename”
 - Type “i” for insert (edit)
 - ESC to go back to entry menu
 - :q – exit without saving
 - :wq – save and exit
 - :q! - force exit without saving

Further Readings

- Linux tutorials:
 - <https://www.tutorialspoint.com/unix/>
 - <https://ryanstutorials.net/linuxtutorial/>
- SLURM – batch system
 - <https://slurm.schedmd.com/documentation.html>
 - <https://doc.itc.rwth-aachen.de/display/CC/Using+the+SLURM+Batch+System>
 - https://hpc-wiki.info/hpc/SLURM#Jobscript_Examples