LIÈGE université
GIGA

Clusters and parallel computing

GIGA doctoral school 2021

Alice Mayer, PhD
GIGA bioinformatic team
bioinfo.giga@uliege.be

Clusters and parallel computing

- Basic notions
- When to use a cluster ?
- Which clusters are accessible to ULG/GIGA members ?
- How to use them ?
- Where to find more information ?

LIÈGE université
GIGA

Basic notions

High Performance Computing (HPC)

Definition

Computing system with extremely high computational power that is able to solve hugely complex problems.

- Analysis of huge volume of data (WGS, high resolution images, etc)
- Compute-intensive processes (simulations, determination of relationship between observations, etc)

	Cores	RAM
My computer	4	16 Gb
GIGA cluster	712	4,456 Gb
Nic5 (CECI)	4,672	20,992 Gb
Juwels (PRACE)	123,408	287,138 Gb

LIÈGE université
GIGA

C.E.C.I
Consortium des Equipements de Calcul Intensif

Partnership for Advanced Computing in Europe

High Performance Computing (HPC)

Definition

Computing system with extremely high computational power that is able to solve hugely complex problems.

- Analysis of huge volume of data (WGS, high resolution images, etc)
- Compute-intensive processes (simulations, determination of relationship between observations, etc)

How to achieve high computational power ?

- Provide powerful machine
- Group several machines together
- Share them and optimize usage

High Performance Microwaving

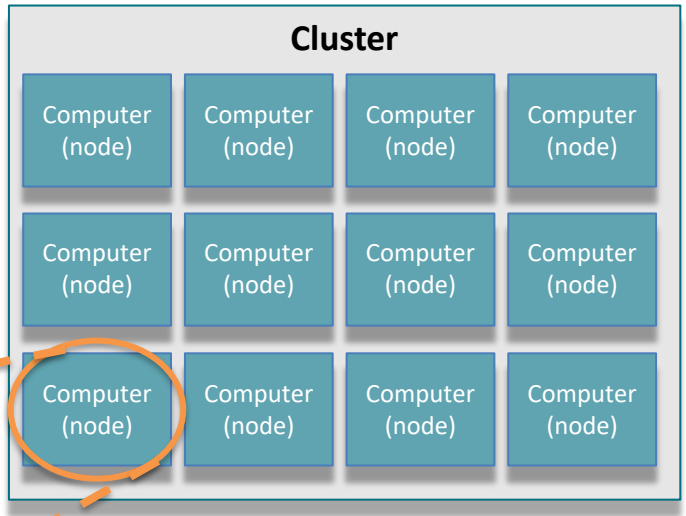
High Performance Computing (HPC)

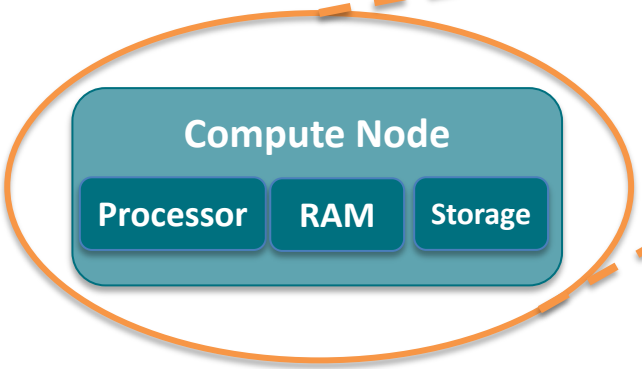
Cluster

Group of linked computers, working together closely so that in many respects they form a single computer

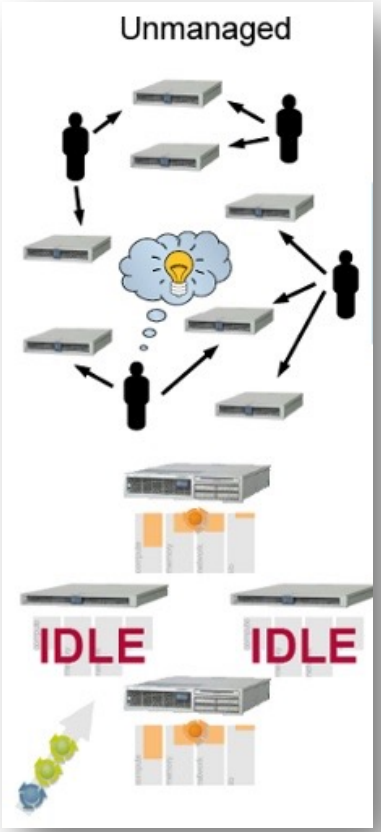
Node

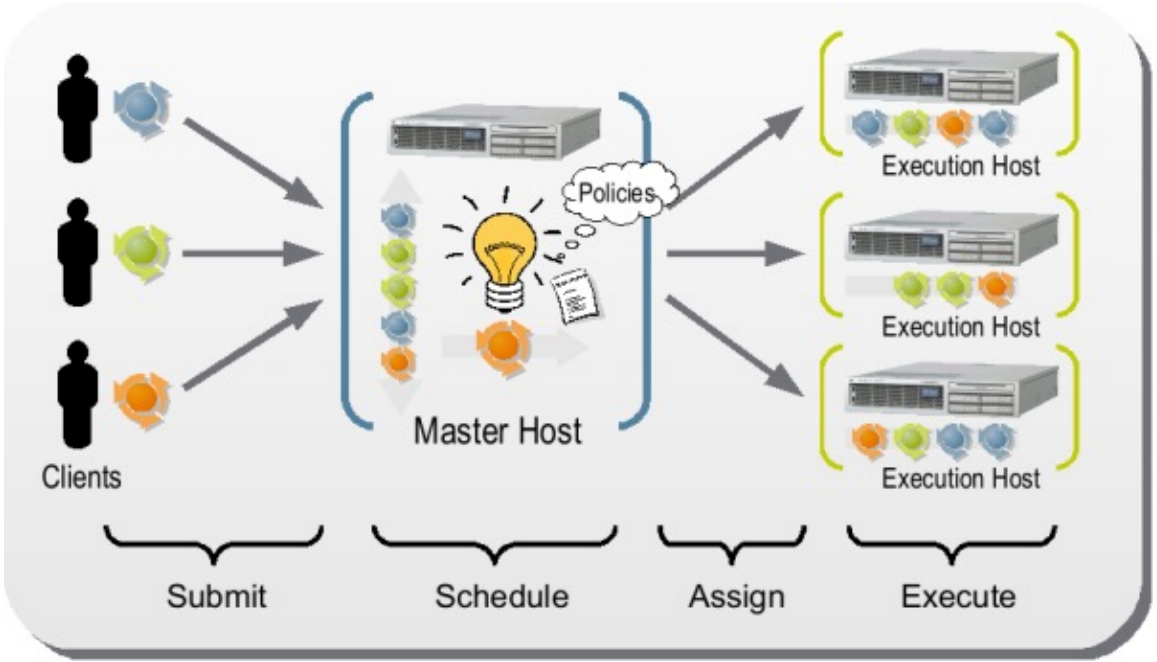
Part of a cluster (equivalent to a high-end workstation)

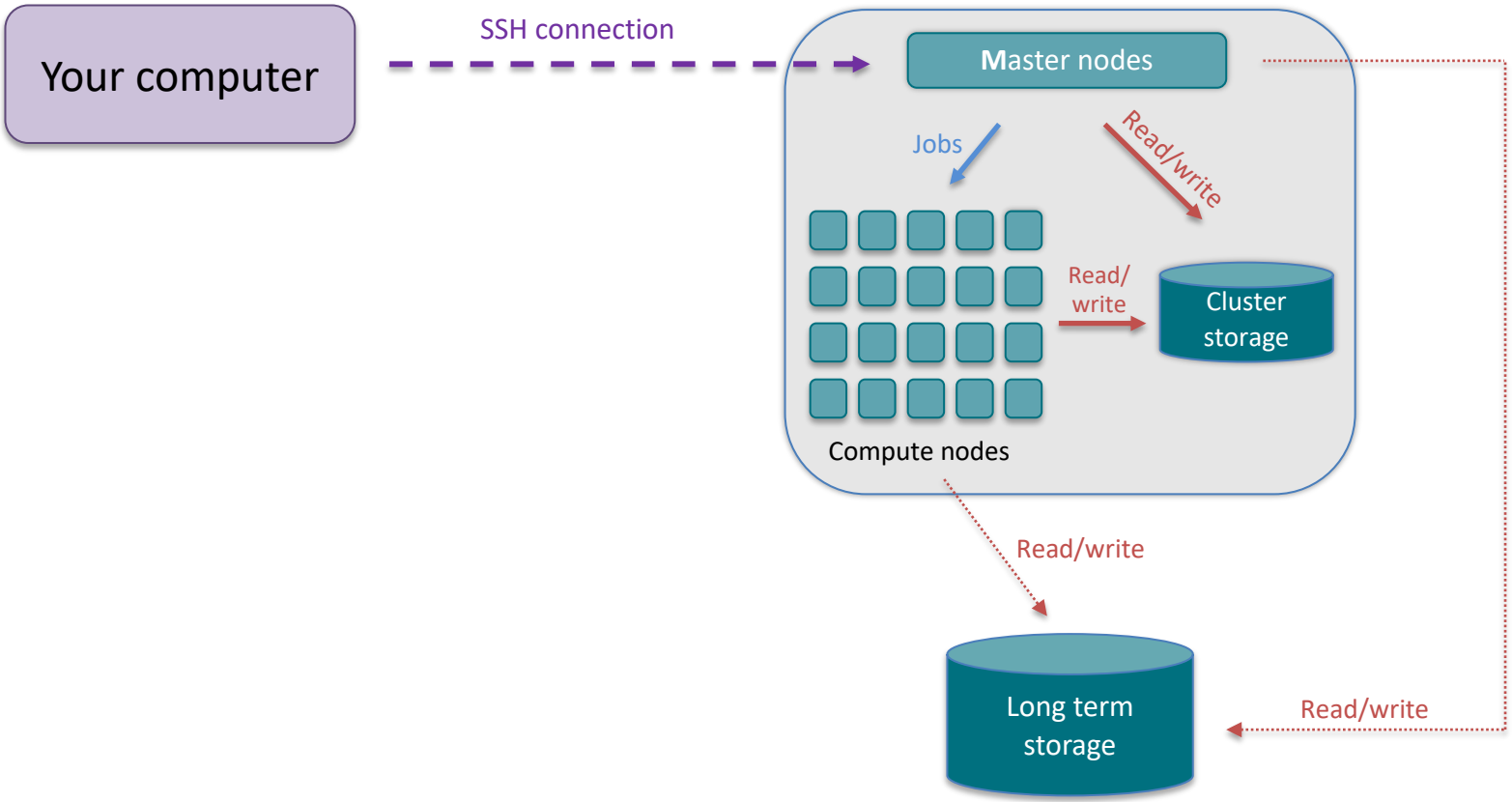




Cluster organisation






Your computer
SSH connection
Master nodes
Jobs
Read/write
Read/
write
Cluster
storage
Compute nodes
Read/write
Long term
storage
Read/write

Why and when do I need to use a cluster ?

When do I need to use a cluster ?

- I need to run the same analysis again and again (on hundreds of samples or testing hundreds values of a given parameter)
- My data don’t fit my disk or my computer’s memory
- The program I use require resources my computer doesn’t have

Use cases : population study

I have to apply the same process to many samples.

Illustration with numbers

- Data from 360 subjects
- Require 1 day/subject/core

Workstation (4 cores) : 90 days

Cluster (720 cores) : 1/2 day

Use cases : whole genome sequencing

My data won't fit in my computer memory.

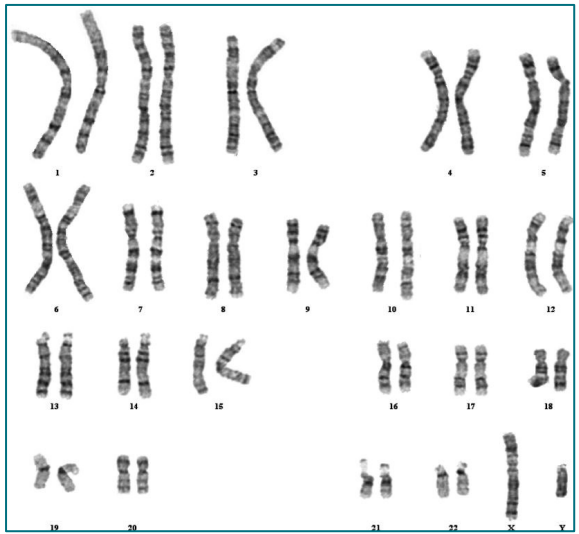


Illustration with numbers

- Human genome = 6 billion bases

  (NB: 6 x 10^9 Seconds = 190 years)
- A single person's whole genome > 300Gb

  and processing it will require > 300Gb RAM
- Of note: in some cases, analysis could be

  split by chromosome and parallelized

Which clusters do I have access to ?

- CECI cluster
- GIGA cluster

CECI

(Consortium des Equipements de Calcul Intensif)

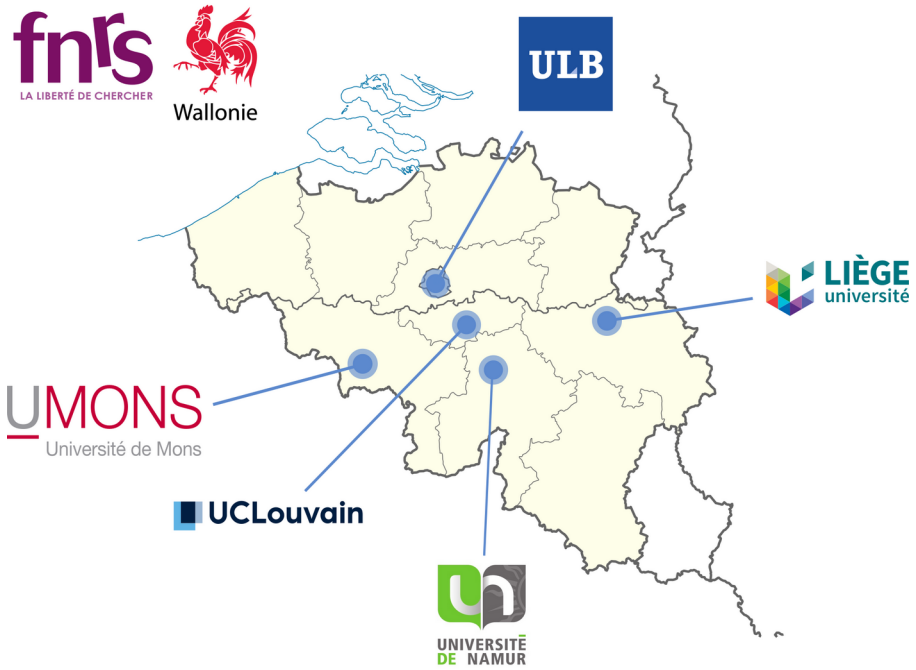
5 universities

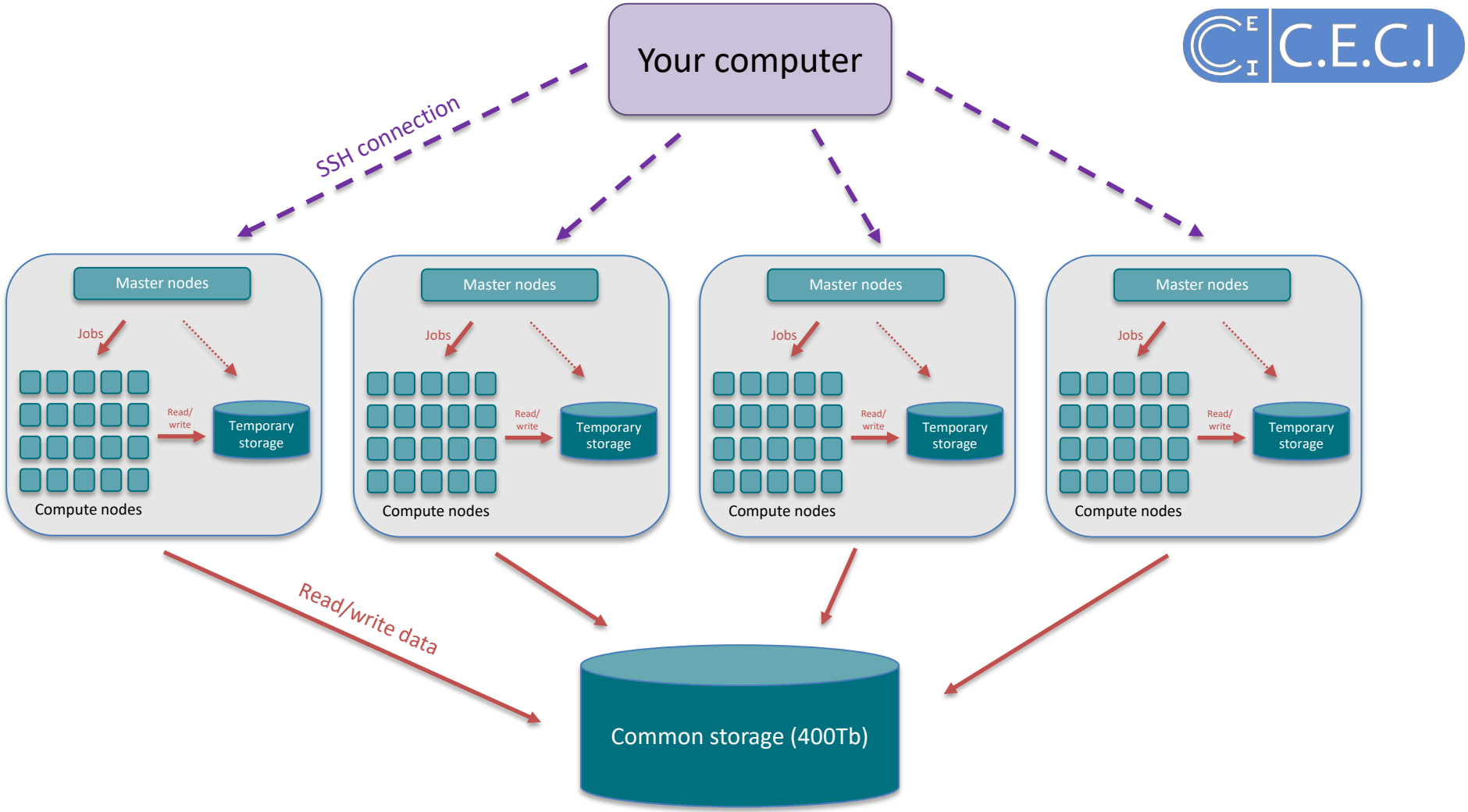
Uliège, UCLouvain, ULB, Umons, UNamur

Website

http://www.ceci-hpc.be/

Support, training, documentation




Your computer
SSH connection
Master nodes
Jobs
Read/
write
Temporary
storage
Compute nodes
Master nodes
Jobs
Read/
write
Temporary
storage
Compute nodes
Master nodes
Jobs
Read/
write
Temporary
storage
Compute nodes
Master nodes
Jobs
Read/
write
Temporary
storage
Compute nodes
Read/write data
Common storage (400Tb)

C.E.C.I

UCLouvain Université

LIÈGE université

ULB

UNIVERSITÉ DE NAMUR

UMONS

Lemaitre 3
2008 cores
Skylake
Haswell

95 GB RAM

Omnipath

Q2 2018

NIC5
4672 cores
AMD Epyc Rome

70*256 GB RAM
3*1 TB RAM

100Gps IB

Vega
2112 cores
Bulldozer

256 GB RAM

QDR IB

Hercules 2
1536 cores
Sandybridge
Epyc

2 TB RAM

10 GbE

Q3 2019

Dragon 2
592 cores
Skylake
Tesla V100

384 GB RAM

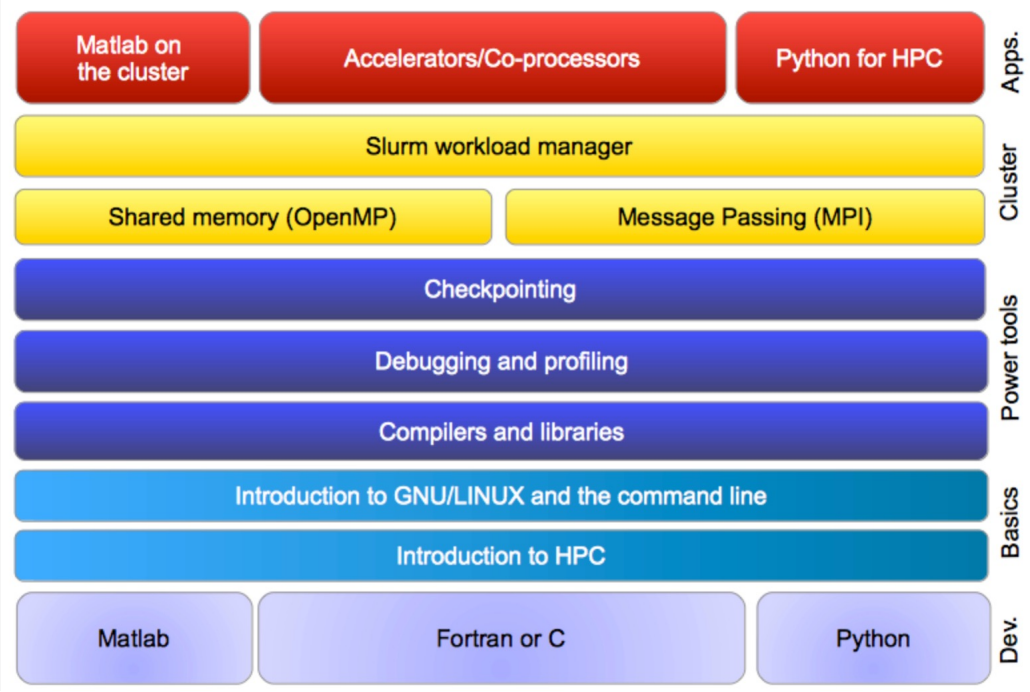
10 GbE

Q1 2019

CECI website and training

http://www.ceci-hpc.be/

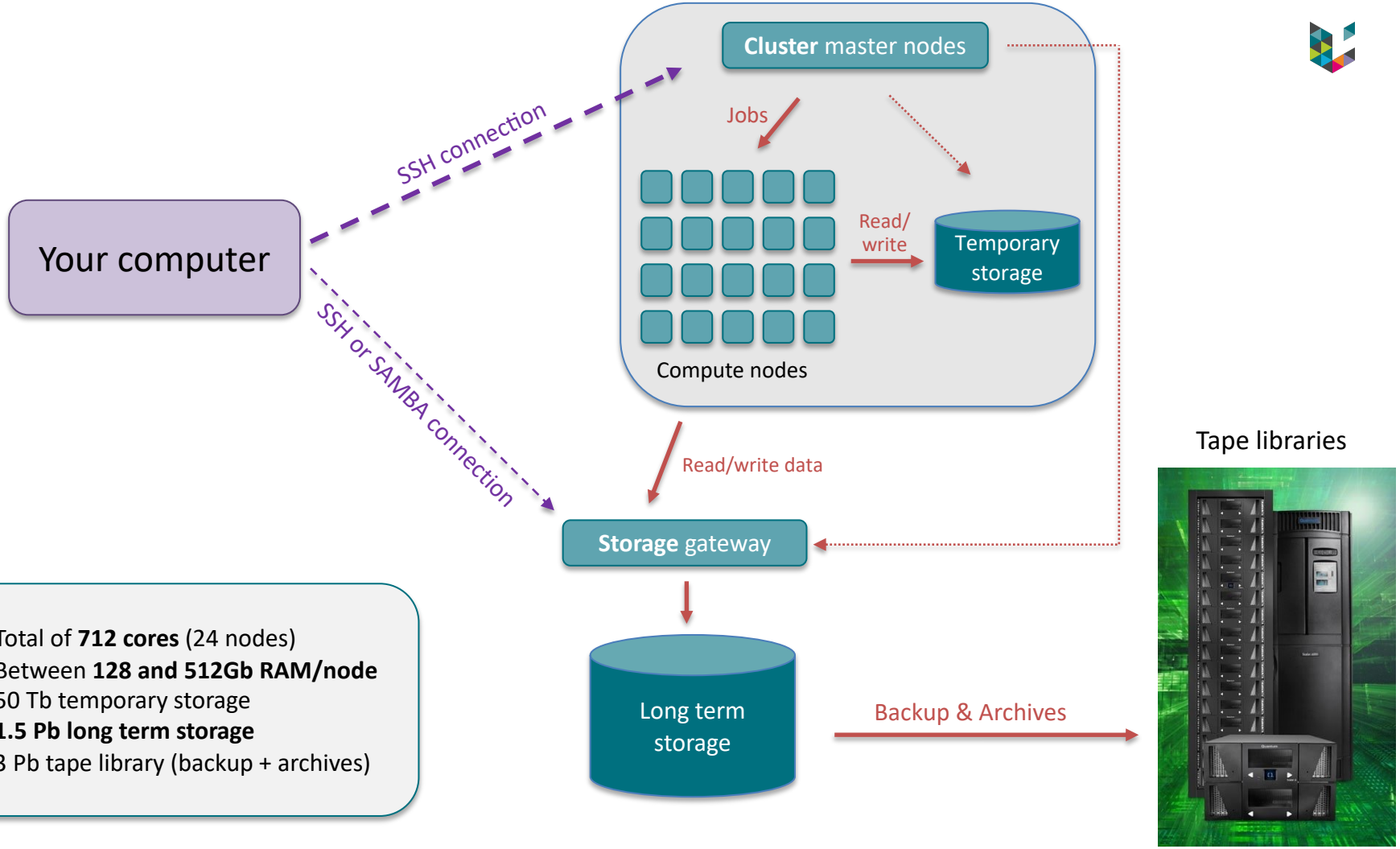
- Support
- Training (18th Oct - 24th Nov)
- Documentation



LIÈGE université
GIGA cluster

- Accessible to all GIGA members and CHU bioinformatic team
- Directly linked to the GIGA mass storage (1.5 petabyte)
- Documentation (work in progress):

  https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/-/wikis/cluster/cluster-home



- Total of **712 cores** (24 nodes)
- Between **128 and 512Gb RAM/node**
- 50 Tb temporary storage
- **1.5 Pb long term storage**
- 3 Pb tape library (backup + archives)

How to use the GIGA or CECI clusters ?

The interface between the user and the cluster: the command line terminal

All CECI and GIGA clusters use CentOS (Linux)

Linux™ CentOS

From MAC and Linux

Terminal

Connection to cluster from a Windows computer

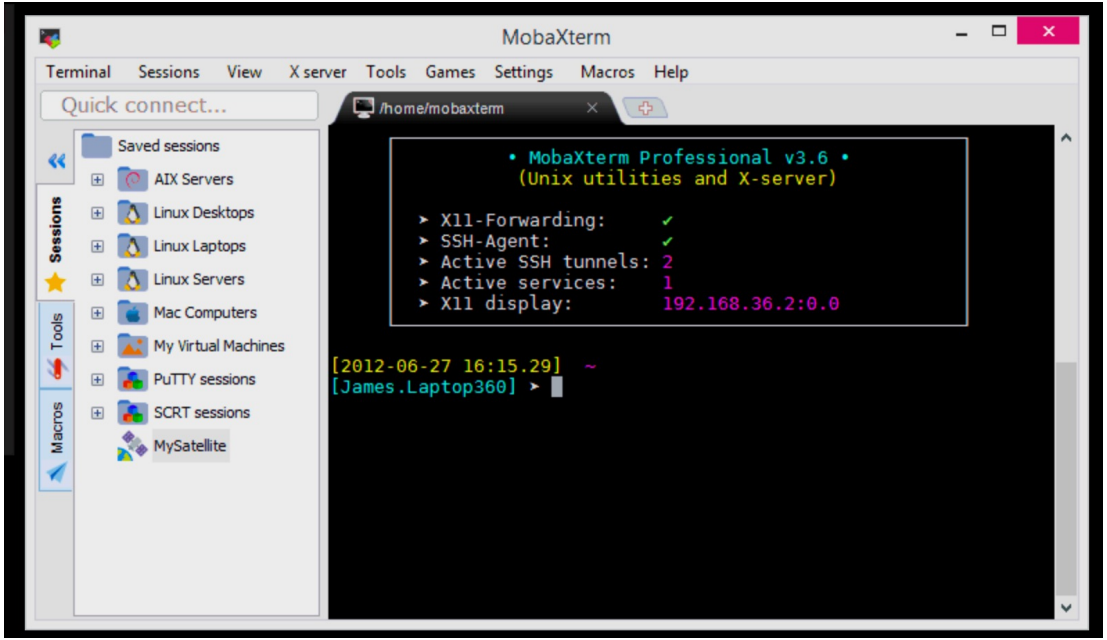
Windows SSH clients

PowerShell
- on Windows 10 and higher,
- looks for it in start menu

MobaXterm
- download from https://mobaxterm.mobatek.net/
- easy to use
- command line interface + interface for file transfer + allow use of graphical applications remotely.

MobaXterm

MobaXterm (recommended by CECI)



How do I connect to the CECI or GIGA cluster ?

CECI cluster

1. Get a CECI account: https://login.ceci-hpc.be/init/
2. Connection instructions: https://support.ceci-hpc.be/doc/_contents/QuickStart

GIGA cluster

Connection instructions (GIGA members):
https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/wikis/mass-storage/mass-storage-connection
(The very first time, it's mandatory to connect to mass storage using SAMBA protocol)

In both cases, if you are outside of university network:

https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/wikis/vpn-connection

How to connect to the GIGA cluster ?

Compared to the mass storage (2 weeks ago)

- SSH only (no SAMBA connection to cluster)
- cluster address instead of mass storage one

Hands-on

1. **Open command line terminal**
   - MAC or Linux : open terminal
   - Windows: Powershell or MobaXterm
2. Type "**ssh u123456@cluster.calc.priv**"
3. (optional) answer yes to message about ECDSA key fingerprint
4. Enter password when prompted

Once logged


alice — u230707@master01:~ — ssh u230707@cluster.calc.priv — 97×26
Last login: Thu Oct 14 15:53:12 on ttys000
alice@ ~ % ssh u230707@cluster.calc.priv
u230707@cluster.calc.priv's password:
Last login: Tue Oct 5 16:55:59 2021 from 10.22.49.17
Welcome to
In case of problem, contact the Helpdesk
Ticket : https://sam.segi.uliege.be/
Phone : 04/366.49.99
E-mail : helpdesk@segi.ulg.ac.be
--> For more information about the GIGA cluster and mass storage:
https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/-/wikis/cluster/cluster-home
u230707@genetic.master01 ~ $

Once logged

Compared to the mass storage (2 weeks ago)

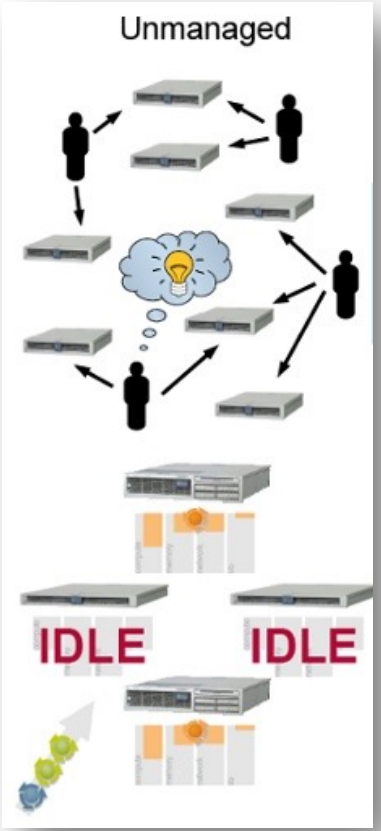
- You are logged into the cluster's master node
- You are in your $HOME (the same as on the mass storage)

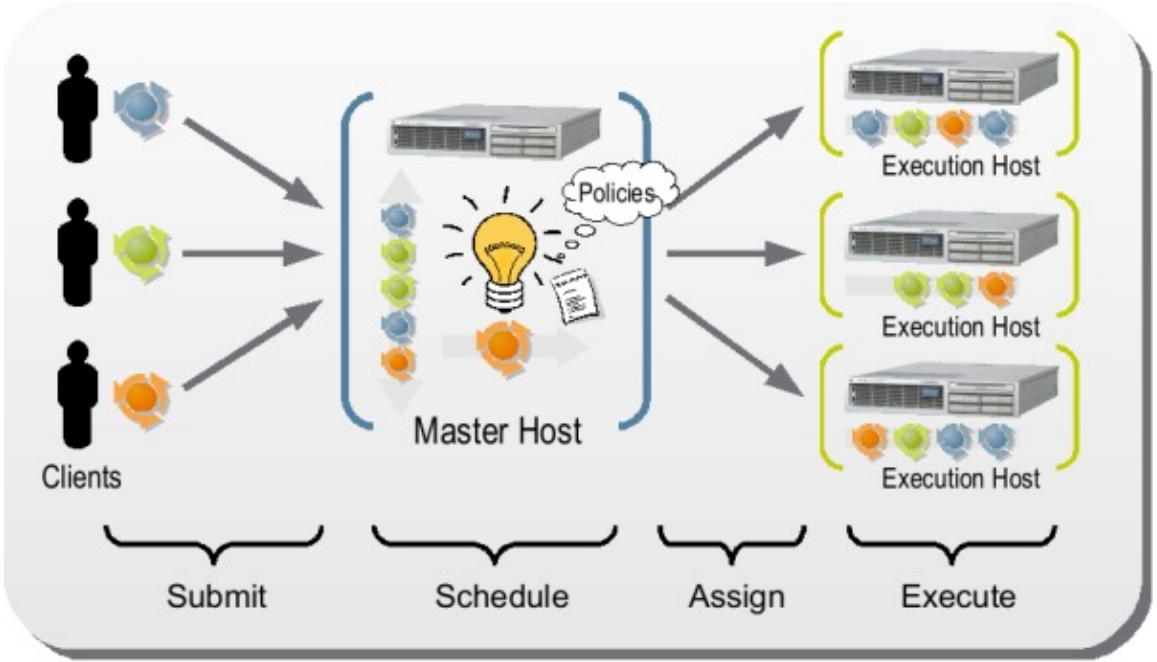
Hands-on

Don't do heavy calculation here but you can try simple bash commands:

- list directory content with "**ls -lh <path>**"
- move around with "**cd <path>**"
- go up one folder (parent folder) with "**cd ..**"
- go back to home with "**cd $HOME**"
- print working directory with "**pwd**" or "**realpath ./**"
- read a text file with "**less <path/to/file.txt>**" (type "**q**" to close it)

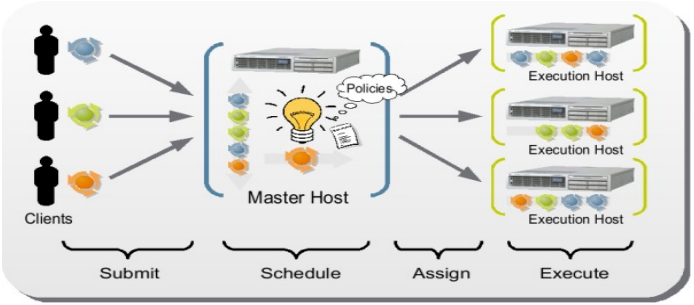
Now that I'm connected, how do I run an analysis?





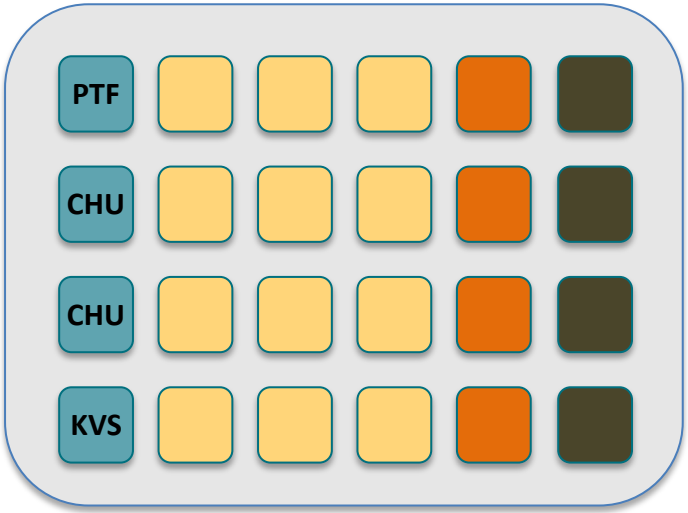
LIÈGE université
GIGA

SLURM
(job scheduler)



Nodes partitions

- Available to all with different time limit (all_5hrs, all_24hrs, kosmos)
- Restricted to a group of users (chugen, ptfgen, urtgen)



all_5hrs

all_5hrs + all_24hrs

all_5hrs + kosmos (infinite)

PTF restricted to 1 team

Slurm basics

GIGA cluster partitions

- Available to all with different time limit (all_5hrs, all_24hrs, kosmos)
- Restricted to a group of users (chugen, ptfgen, urtgen)

Before using slurm

```
$ module load slurm
```

Getting info about nodes

```
$ sinfo
$ cat /etc/slurm/slurm.conf | grep ^Node
$ squeue
```

```
u230707@genetic.master01 ~ $ cat /etc/slurm/slurm.conf | grep -i ^Node
NodeName=chugen001 CoresPerSocket=10 RealMemory=128000 Sockets=2 ThreadsPerCore=1 TmpDisk=200000 Weight=100 Feature=intel
```

RealMemory

RAM in Mb (ex: 128 Gb for chugen001)

CPU (or socket)

Central Processing Unit that contains one or several core(s) + other components

Core

Independent processing unit that reads and executes instructions of a program

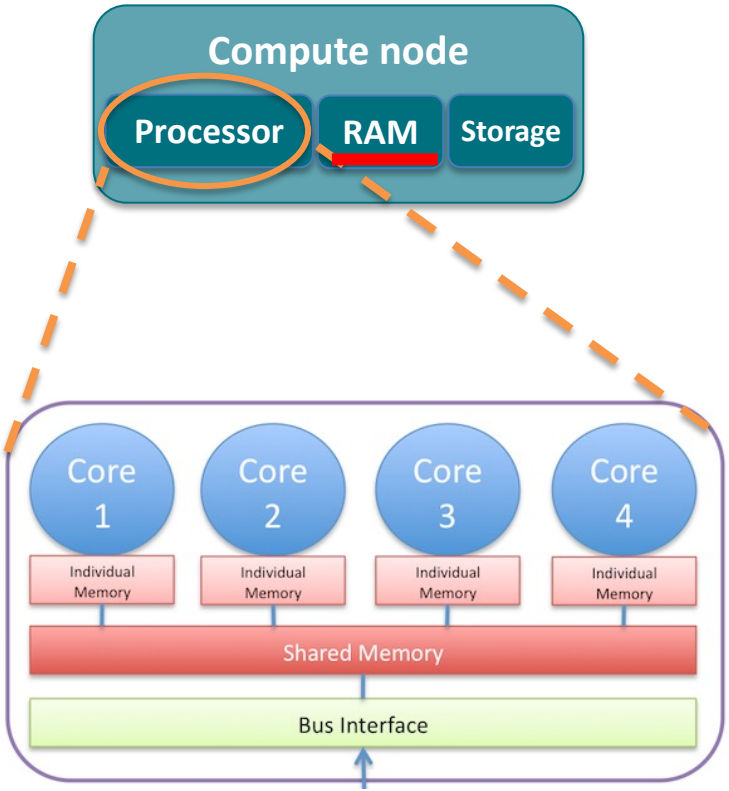


image from https://www.cse.wustl.edu/~jain/cse567-11/ftp/multcore/

Two types of slurm sessions

Interactive sessions

- Several short tasks
- Tasks that require user input
- Typically: when developing/optimizing pipeline

Batch sessions

- Longer running processes
- Parallel processes

Slurm interactive sessions

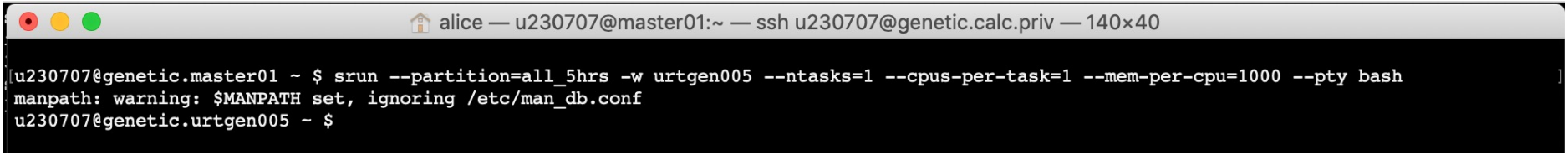
How to start one ?

$ srun --partition=all_5hrs --cpus-per-task=1 --mem-per-cpu=1000 --pty bash

- Asking for 1 core and 1 Gb of RAM on a node of the all_5hrs partition
- Asking to have bash session on the allocated node

Notice the change of prompt, from **u230707@genetic.master01** to **u230707@genetic.urtgen005** !!!!

Slurm interactive sessions



Slurm interactive session (srun)

You are now on a node

You can perform analysis there

If you use more resources than requested, slurm will kill your session on the node

If you lose your internet connection, your session will be aborted, and your program will crash

Slurm interactive sessions

Monitor jobs

while job is still running, give info on resources, nodes, etc
$ scontrol show job <JOB_ID>

After job finished, info on resources used
$ sacct --format="JobId,JobName,NodeList,State,Elapsed,CPUTime,MaxRSS,AveRSS,ReqMem, ReqCPUS, Submit,Start" -j <JOB_ID>

Don’t forget to close it when you’ve finished !!!!

$ exit

Batch jobs

myscript.sh

Resources requested (http://www.ceci-hpc.be/scriptgen)	#!/bin/bash # #SBATCH --ntasks=1 #SBATCH --cpus-per-task=1 #SBATCH –mem-per-cpu=1000 #SBATCH –partition=all_5hrs #SBATCH --time=1:00:00 #SBATCH --mail-user=my@email.com #SBATCH --mail-type=FAIL
Instructions (Shell script, Python...)	# Do some stuff echo “Hello”

submit

```
$ sbatch myscript.sh
```

```
#!/bin/bash
#
#SBATCH --job-name=Test
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=1000    # in Mb (could also write 1G)
#SBATCH --time=1:00:00    # "hours:minutes:seconds"
#SBATCH --partition=all_5hrs
#SBATCH --output=test_%j.log   # path + name of log file, %j = job ID
#SBATCH --mail-user=alice.mayer@uliege.be
#SBATCH --mail-type=FAIL

date
# Run stuff here
echo "Hello" > hello.txt
sleep 120     # do nothing during 2 minutes

#########################################################
### Printing out info about slurm job   ##############
#########################################################
echo ""
echo "scontrol show job ${SLURM_JOB_ID} output:"
echo ""
scontrol show job ${SLURM_JOB_ID}
echo ""
date
```

1. Write and save the script as **test.sh**
   - Copy/paste from last slide
   - Change user email address
2. Launch it with "**sbatch test.sh**"
3. Monitor with **squeue**
4. Once finished, check log file

Batch jobs

Monitor jobs

while job is still running, give info on resources, nodes, etc
$ scontrol show job <JOB_ID>

After job finished, info on resources used
$ sacct --format="JobId,JobName,NodeList,State,Elapsed,CPUTime,MaxRSS,AveRSS,ReqMem, ReqCPUS, Submit,Start" -j <JOB_ID>

Cancel jobs

$ scancel <jobID>

Where to find programs on the cluster?

System defaults

Some programs are available "by default" (ex: Python 2.7.5)

Modules (managed by sys-admin)

Centralised installation of commonly used tools

$ module load EasyBuild
$ module avail
$ module avail <ModuleName> # case sensitive!!!!
$ module load <ModuleName> # to load the module and use the program

Installed by bioinformatic team

In **$HOME/_SHARE_/Resources/Tools** (singularity containers + nf-core pipelines + softwares)

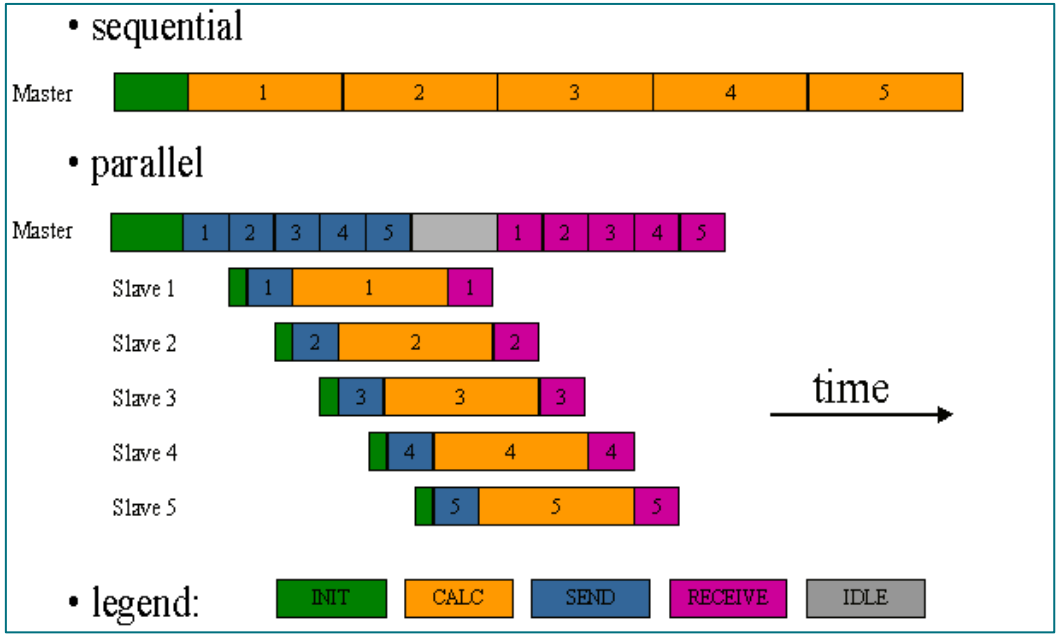
"**module avail**" is case sensitive !!!

```
u230707@genetic.master01 ~ $ module load EasyBuild
u230707@genetic.master01 ~ $ module avail python

------------------------------------------------------------ /cm/shared/modulefiles -------------------------------------------------------------
python/3.7
u230707@genetic.master01 ~ $ module avail Python

------------------------------------------- /cm/shared/apps/easyb/.local/easybuild/modules/all --------------------------------------
Python/2.7.11-goolf-1.7.20       Python/2.7.14-GCCcore-6.4.0-bare Python/2.7.16-GCCcore-8.3.0      Python/3.7.4-GCCcore-8.3.0
Python/2.7.12-foss-2016b         Python/2.7.15-foss-2018b         Python/2.7.18-GCCcore-9.3.0      Python/3.8.2-GCCcore-9.3.0
Python/2.7.14-foss-2017b         Python/2.7.15-GCCcore-7.3.0-bare Python/3.6.6-foss-2018b
u230707@genetic.master01 ~ $
```

LIÈGE université
GIGA

Parrallel processing

How much can I parallelize in practice ?



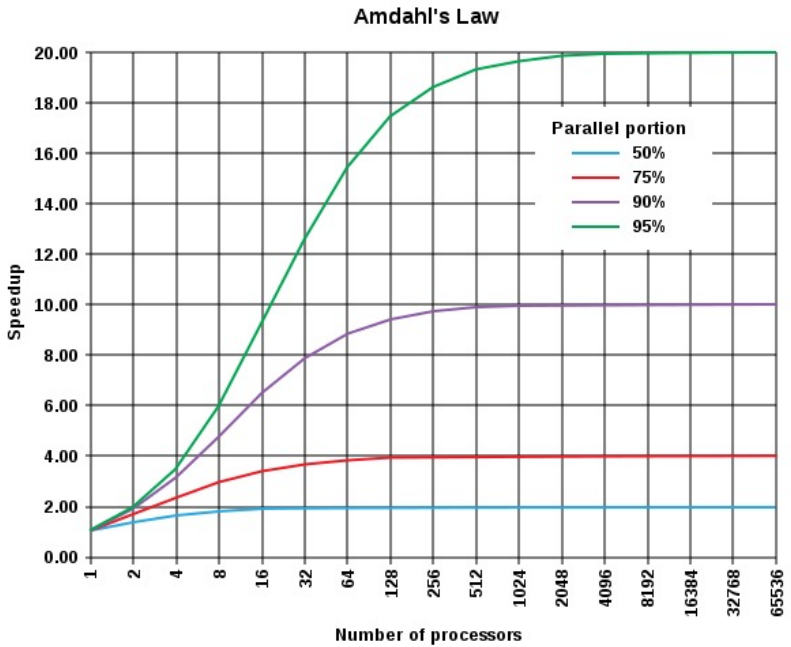
- Worth if time(subtask) >> time(overheads)
- Time saved α fraction parallelizable (Amdahl's law)

Amdahl's law

$$\text{Speedup} = \frac{1}{(1-F)+\frac{F}{N}}$$

F = parallelisable fraction
N = number of nodes
Assume no overhead for

- Scheduling
- Networking
- Synchronisation



Using slurm array to parallelize

This script will launch 4 jobs (by 2). Each one will write its number in the log and wait 2 minutes.

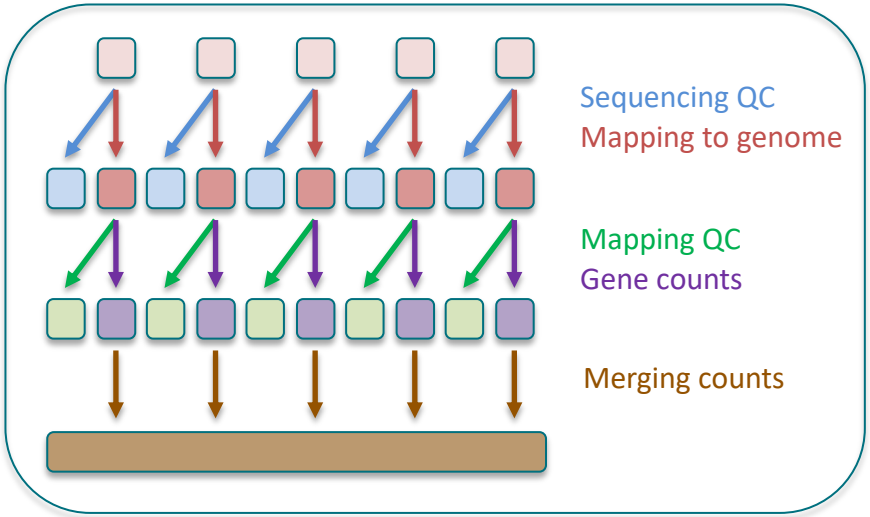
```
#!/bin/bash
#
#SBATCH --job-name=Test
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=1000    # in Mb (could also write 1G)
#SBATCH --time=1:00:00    # "hours:minutes:seconds"
#SBATCH --partition=all_5hrs
#SBATCH --output=test_%j.log   # path + name of log file, %j = job ID
#SBATCH --mail-user=alice.mayer@uliege.be
#SBATCH --mail-type=FAIL
#SBATCH --array=1-4%2

date
# This will be printed in the log file of each job
echo ""
echo "Hello, I'm the job number ${SLURM_ARRAY_TASK_ID}"
sleep 120     # do nothing during 2 minutes
date
```

1. Write and save the script as **test_array.sh**
   - Copy/paste from last slide
   - Change user email address
2. Launch it with "**sbatch test_array.sh**"
3. Monitor with **squeue**
4. Once finished, check log file

Workload manager (ex: nextflow)

Tools developed to process several samples through several analysis steps,
while optimizing resources usage

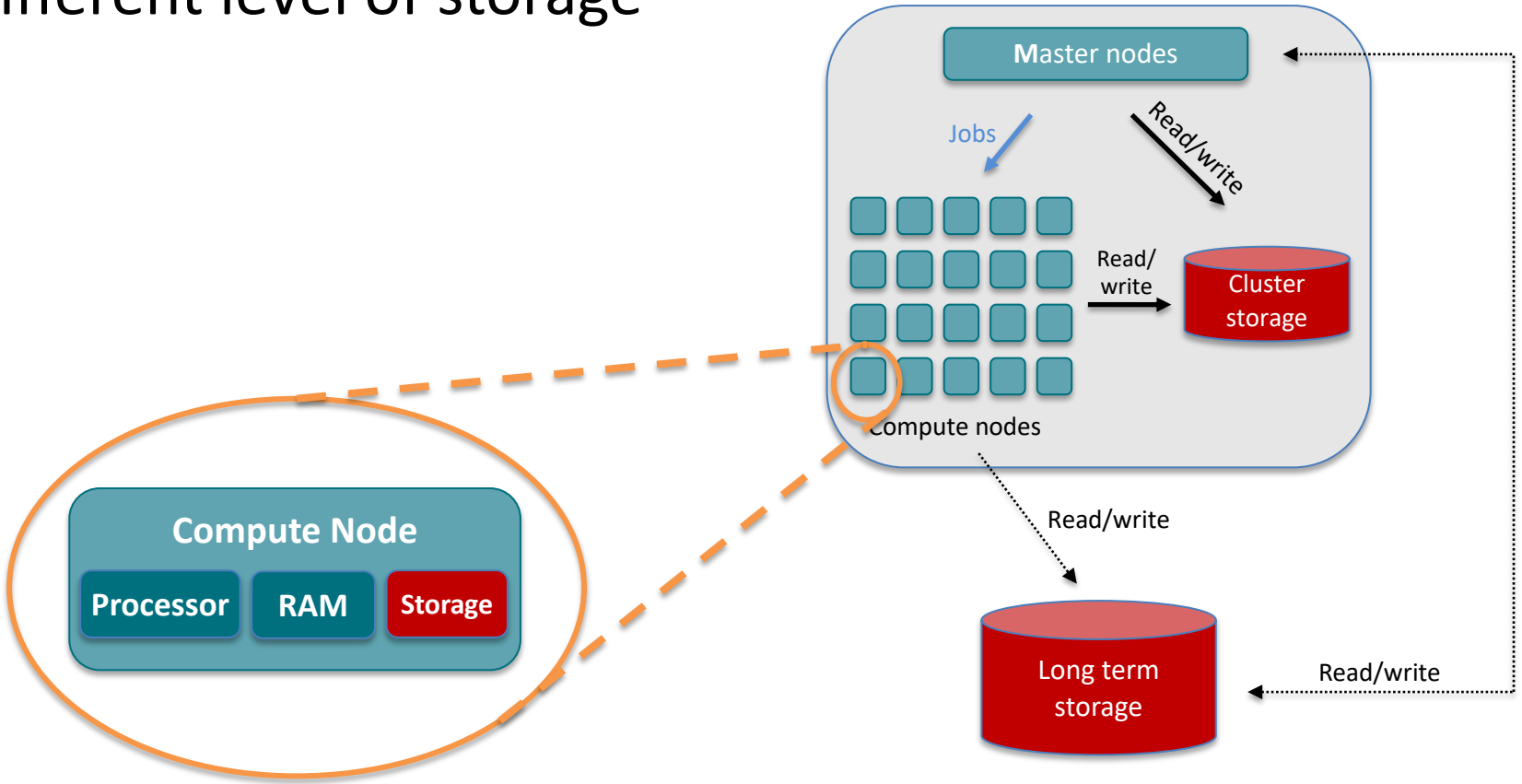
nf-core/rnaseq



https://nf-co.re/rnaseq/output
https://www.nextflow.io/docs/latest/tracing.html

Data managment

Different level of storage



Different level of storage on GIGA cluster

	Path from node	Speed	Space	Accessible	Backup
Node storage	/local	++	**2 Tb**	Only from 1 node	no
Cluster storage	/gallia/scratch	+	**50 Tb**	From all nodes	no
Mass storage	/massstorage	-	**1500 Tb**	From all nodes	yes

At the end of your job, don't forget to transfer and delete

- all your files from node storage
- everything you won't need anymore from the cluster storage

Different level of storage on CECI clusters

	Path from node	Speed	Space	Accessible	Backup
Node storage	Local scratch	+++	-	Only from 1 node	no
Cluster storage	Workdir	++	+	From all nodes of 1 cluster	no
	Home	+			
Global storage	GlobalHome + Transfer	-	+++	From all nodes and clusters	no

For more information: https://support.ceci-hpc.be/doc/_contents/ManagingFiles/Storage.html

WARNING: Data in the Workdir can be removed at any time especially during maintenance periods.

Take-Home message and useful links

Take-home messages

- Cluster = group of powerful compute nodes linked together
- Clusters are very useful when an analysis is not possible or too slow on our desktop computer
- When using a cluster,
  - don’t calculate on master node but use slurm to send jobs to compute nodes instead
  - write temporary and intermediate files on node or cluster temporary storage and not directly on mass storage
- When your analysis is finished
  - Transfer final output to long term storage
  - Delete temporary and intermediate files from node and cluster storage

Useful links

CECI clusters

- CECI website: http://www.ceci-hpc.be/
- documentation: https://support.ceci-hpc.be/doc/ (including slurm tutorial and FAQ)
- Training: http://www.ceci-hpc.be/training.html (session starting today, including "Efficient use of Matlab on the cluster" on 25th November)

GIGA clusters

- wiki: https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/wikis/home
- slurm page: https://gitlab.uliege.be/giga-bioinfo/user-guides-wiki/wikis/cluster/slurm/slurm_home
- slurm manual: https://slurm.schedmd.com/archive/slurm-14.11.11/quickstart.html
- Contact: https://sam.med.uliege.be/ (choose UDI-MED or BIOINFO-GIGA as category)

LIÈGE université
GIGA

Thank you for your attention !
Questions ?

Alice Mayer, PhD
GIGA bioinformatic team
bioinfo.giga@uliege.be


LIÈGE université
GIGA

HPC in Europe

Definition

Computing systems with extremely high computational power that are able to solve hugely complex and demanding problems.

EU priority

HPC is one of the key digital domains where **the EU's investment is due to significantly increase** […]. Moreover, supercomputing will play a key role in Europe's path towards recovery, as it has been identified a **strategic investment priority**.

https://ec.europa.eu/digital-single-market/en/high-performance-computing

Applications

- monitoring and mitigating the effects of climate change
- producing safer and greener vehicles
- **advancing the frontiers of knowledge** in nearly every scientific field
- **drug design**, from testing drug candidate molecules to repositioning existing drugs for new diseases
- **understand the origins and evolution of epidemics and diseases**.

Example

Fighting coronavirus: European supercomputers join pharmaceutical companies in hunt for new drugs

High Performance Computing

Evolving concept

VS

Partnership for Advanced Computing in Europe

- 26 member countries
- 7 supercomputers (5 host countries)
- Ex: Juwels (Germany):
  - 287,136 Gb RAM
  - 123,408 cores

https://prace-ri.eu/hpc-access/hpc-systems/