



Vlaanderen  
is supercomputing

# vsc HPC introduction course

<https://hpcleuven.github.io/HPC-intro>



# Training material

- Everything is available on Github:  
<https://hpcleuven.github.io/HPC-intro/>
- Video recordings
  - Scan the QR code
  - Recommended videos: ~2 hrs
  - Optional videos: ~1 hr

By registering and participating in VSC trainings, you accept to abide by the  
[VSC training Code of Conduct](#)



# What is High Performance Computing?

- ❑ Using supercomputers to solve advanced computational problems
- ❑ Reduce the computation time from days, years, decades or centuries to minutes, hours, days or weeks
- ❑ The key is parallelism
- ❑ Access to specialized hardware (GPU, large memory, high-speed interconnect)



# What is High Performance Computing?



The concept is simple: **parallelism** = employing multiple processors for a single problem

# Training outline

## Part 1: Basics

- VSC
- Tier-2 clusters
- Storage
- Accounting
- Login
  - Open OnDemand
  - SSH client & terminal
- Data transfer
  - FileZilla
  - rsync
  - Globus

## Part 2: Hands-on

- Using Open OnDemand
- Software
- Miniconda
- Batch jobs
- Demo: try things yourself



**VSC**  
(Vlaams Supercomputer Centrum)

# VSC PARTNERSHIP



Supported by



Visit <https://vscentrum.be>

Extensive set of courses  
every academic year

Links to the VSC account page and the VSC documentation  
(answers >80% of your questions!)

The screenshot shows the homepage of the VSC website. At the top, there is an orange navigation bar with the text "Visit <https://vscentrum.be>". Below this, the main header features the text "Extensive set of courses every academic year" and "Links to the VSC account page and the VSC documentation (answers >80% of your questions!)". The main content area has a dark background with a light gray central banner. The banner contains the text "Welcome to VSC Flanders' most highly integrated high-performance research computing environment" in large, bold, black font. At the bottom right of the banner is a button labeled "Read More". The footer of the page includes the VSC logo ("VLAAMS SUPERCOMPUTER CENTRUM" with a stylized lion icon) and the text "Vlaanderen is supercomputing". The footer also contains links to "Home", "About VSC", "Systems & Services", "Showcase", "News & Events", "VSC Training", "User Portal", and "Access". A search bar with a magnifying glass icon is located at the top right of the main content area.

VLAAMS  
SUPERCOMPUTER  
CENTRUM

Vlaanderen  
is supercomputing

Home About VSC Systems & Services Showcase News & Events VSC Training User Portal Access

# Welcome to VSC

# Flanders' most highly integrated

# high-performance research

# computing environment

Read More



## Welcome to the User Portal

Here you can find the gateway to the User documentation of the Vlaams Supercomputer Centrum

User documentation page

Manage your VSC-account  
*partner institute account required*

VSC account page

<https://docs.vscenrtum.be>

Welcome to the VSC documentation

The VSC documentation offers extensive *how-to* guides and technical information about the services provided by the [Vlaams Supercomputer Centrum](#).

**Accounts and access**  
How to get your VSC account and access the different VSC services and platforms.

**Research Data**  
Data transfer and storage in the VSC infrastructure.

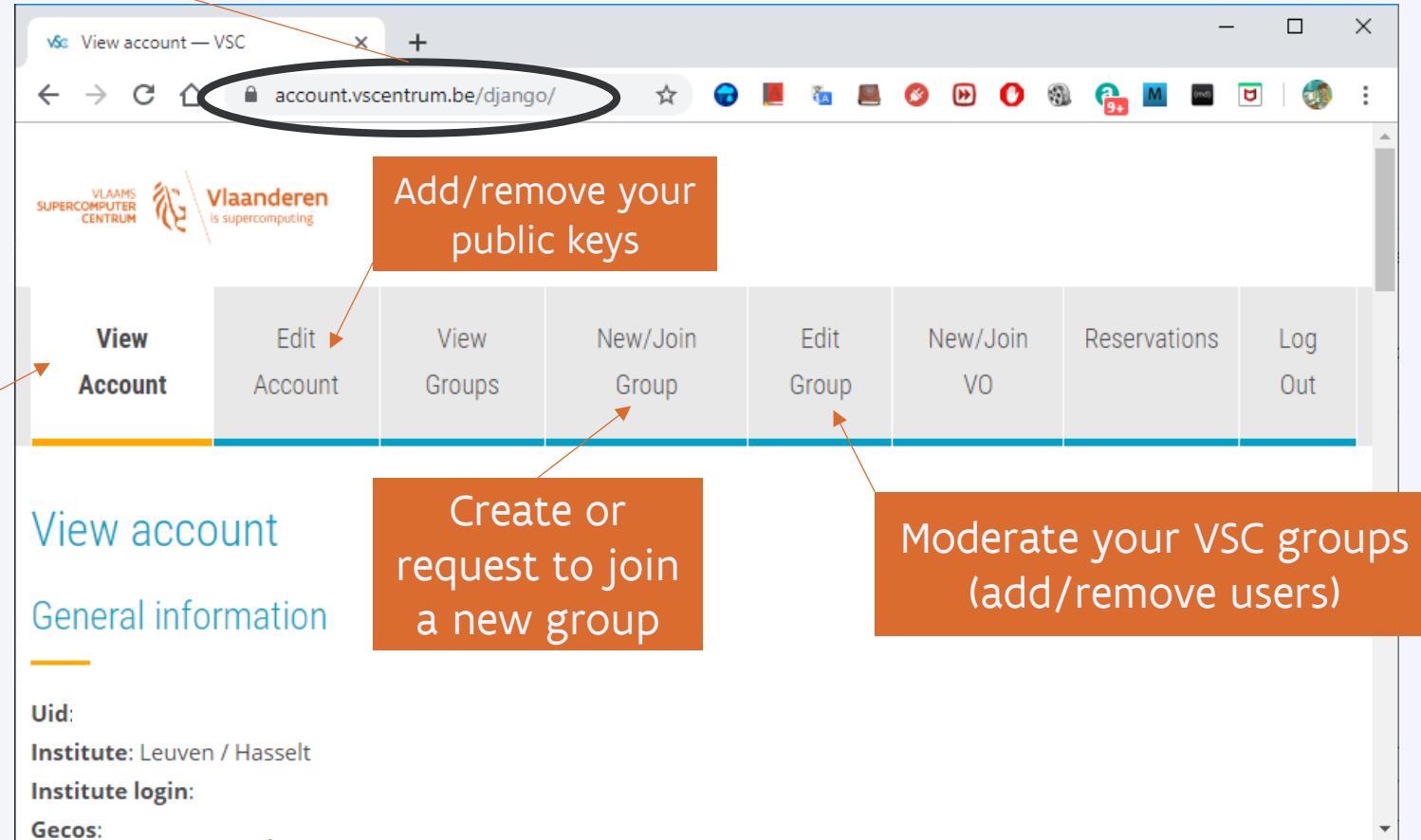
**Compute**  
The high-performance computing (HPC) platform provides multiple tiers of parallel processing

**Tier-1 Cloud**  
The VSC Cloud component provides *on-demand* resources in a more flexible and cloud-like

**Tier-1 Data**  
The VSC Data component enables research data to remain close to the computing infrastructure

Enter your keywords here; e.g. 'Ondemand'

Manage your VSC account at  
<https://account.vscentrum.be>



# Support and services

## Basic support

- Helpdesk ([hpcinfo@kuleuven.be](mailto:hpcinfo@kuleuven.be))
- Monitoring and reporting

## Application support

- Installation and porting
- Optimisation and debugging
- Benchmarking
- Workflows and best practices

## Training

- Documentation and tutorials
- Scheduled trainings and workshops
- Workshops on request
- One-to-one sessions

# Become a VSC user

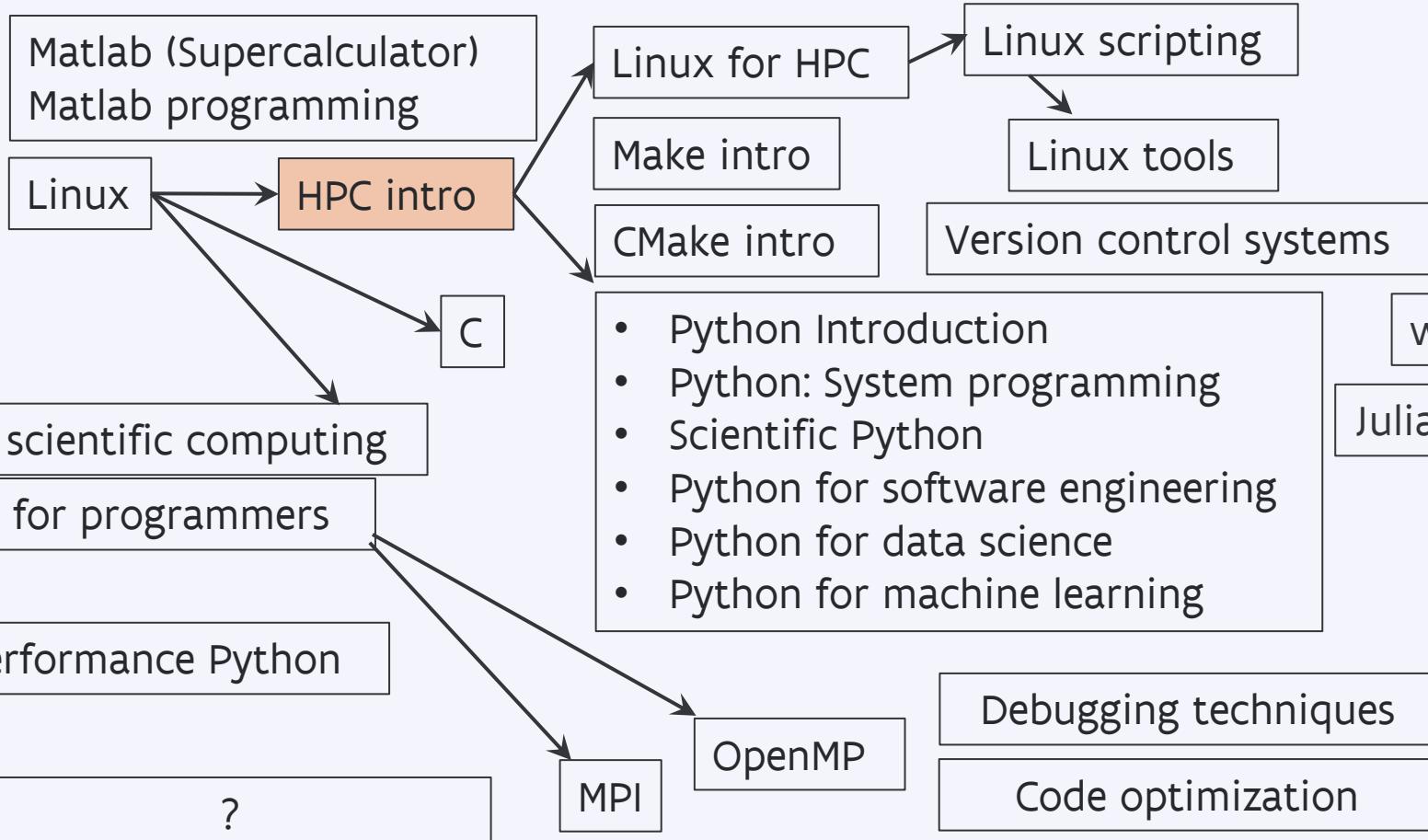
- (Optional) Create a secure (4096 bit) [SSH key pairs](#)  
Upload it on the account page: <https://account.vscentrum.be>
- You need to [request a VSC account](#)  
Normally gets processed swiftly
- Request [introductory credits](#) (2M free credits for 6 months)
- Request [project credits](#) (for supervisors and project leaders)  
You need to create a VSC group  
Add users to the group to give them access to use credits  
Fill out the request form
- Extra storage requests  
Scratch extension: free of charge  
Staging storage: 20 € per TB per year
- All service costs (compute and storage) are described in the ICTS service catalog: <https://icts.kuleuven.be/sc>  
Click on [High Performance Computing](#) (NL/EN)

The screenshot shows a web browser window titled 'View account — VSC'. The URL is 'account.vscentrum.be/djang...'. The page header includes the Vlaams Supercomputer Centrum logo and the text 'Vlaanderen is supercomputing'. A navigation bar at the top has tabs for 'View Account' (which is highlighted in blue), 'Edit Account', 'View Groups', 'New/Join Group', 'Edit Group', 'New/Join VO', 'Reservations', and 'Log Out'. Below the navigation bar, the main content area is titled 'View account' and 'General information'.

The screenshot shows a web browser window titled 'ICTS Servicecatalogus'. The URL is 'icts.kuleuven.be/sc'. The page header includes the KU Leuven logo and the text 'ICTS SERVICECATALOGUS'. It features a 'MENU' button, language links ('NL EN'), and a user profile ('Ehsan Moravveji'). The main content area includes sections for 'Home', 'ICTS SERVICECATALOGUS', 'WAT IS DE ICTS SERVICECATALOGUS?', 'OPGELET', and 'ZOEKEN'.

# VSC training

- Introductory



- Intermediate

- Advanced

- Specialized track

Infosessions:  
• Containers  
• Notebooks

Stay up-to-date <https://www.vscentrum.be/en/education-and-trainings>

# How to acknowledge VSC in publications

## Why?

- a contractual obligation for the VSC
- helps VSC to secure funding
- you will benefit from it in the long run

## For KU Leuven users

- add the relevant papers to the “KUL-HPC” virtual collection in Lirias

## In het Nederlands

*De infrastructuur en dienstverlening gebruikt in dit werk werd voorzien door het VSC (Vlaams Supercomputer Centrum), gefinancierd door het FWO en de Vlaamse overheid.*

## In English

*The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government.*



## Tier-2 clusters

# VSC HPC infrastructure



# Tier-2 clusters @ KU Leuven

Genius (since 2018)

250 nodes | 8936 cores



Most of Genius will be decommissioned in early 2026

wlCE (since September 2022)

186 nodes | 13392 cores



# Tier-2 cluster – Genius

Node type	CPU type	# cores	Total mem	Mem per core (MB)	Partition	Planned end of life
Cascadelake	Xeon Gold 6240	36	192 GB	5000	batch	Q1 2026
Skylake large mem	Xeon Gold 6240	36	768 GB	21000	bigmem	Q1 2026
Skylake P100 GPU	Xeon Gold 6140 4xP100 SXM2 16GB	36	192 GB	5000	gpu_p100	2027 (?)
Cascadelake V100 GPU	Xeon Gold 6240 <b>8xV100 SXM2</b> 32GB	36	<b>768 GB</b>	<b>21000</b>	gpu_v100	2027 (?)
Superdome	Xeon Gold 6132 8 sockets	112	<b>6 TB</b>	<b>53500</b>	superdome	Q1 2026
Skylake interactive	Xeon Gold 6140	36	<b>192 GB</b>	<b>5000</b>	interactive	Q1 2026

- Access: login.hpc.kuleuven.be
- Interconnect: InfiniBand EDR (25 Gb/s)
- SSD Disks: 200 GB

- Interactive partition:  
max resources: 8 cores, 16hr

Remarks

# Tier-2 cluster – wICE

Node type	CPU type	# cores	Total mem	Mem per core (MB)	Partition
Icelake	Xeon 8360Y	72	256 GB	3455	batch_icelake
Sapphire Rapids	Xeon 8468	96	256 GB	2500	batch_sapphirerapids
Icelake large mem	Xeon 8360Y	72	2 TB	28000	bigmem
Icelake large mem	Xeon 8360Y	72	8 TB	111900	hugemem
Icelake A100 GPU	Xeon 8360Y 4xA100 SXM2 80GB	72	512 GB	7000	gpu_a100
Genoa H100 GPU	Epyc 9334 4xH100 SXM5 80GB	64	768 GB	11700	gpu_h100
Icelake interactive	Xeon 8358 1xA100 SXM2 80GB	64	512 GB	7500	interactive

- Phase 1: since September 2022
- Access: login.hpc.kuleuven.be
- Interconnect: InfiniBand HDR (100 Gb/s)

- SSD Disks: 960 GB
- Interactive partition:  
max resources: 8 cores, 1 GPU, 16hr

Remarks

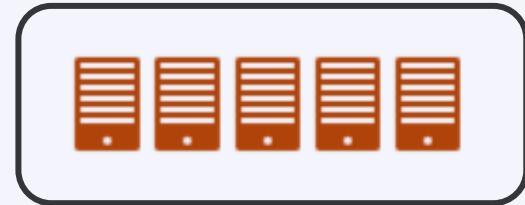
# Tier-2 cluster – wICE

Thin nodes



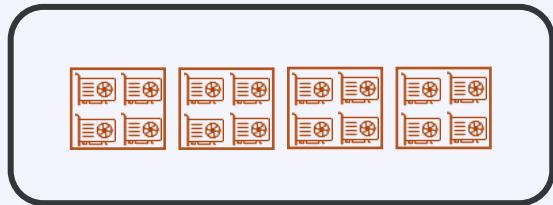
172x IceLake 72c 256 GB

Large memory nodes



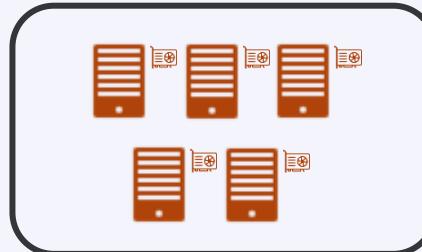
5x IceLake 72c 2048 GB

GPU nodes



4x IceLake 72c 512 GB  
4 A100 SXM4 80GB

Interactive nodes



4x IceLake 64c 512 GB  
1 A100 80GB

No Dedicated  
Login Nodes

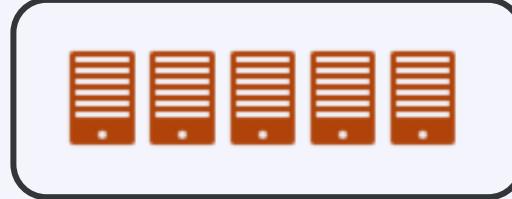
# Tier-2 cluster – wICE extension

Thin nodes



68x Sapphire Rapids 96c 256 GB

Large memory nodes



1x IceLake 72c 8 TB

GPU nodes



4x AMD Genoa 64c 768 GB  
4x H100 SXM5 80GB

# Storage

# Overview of the Tier-2 storage locations

- ✓ Only you own your files (POSIX)  
Users can share folders via [VSC groups](#)
- ✓ A VSC account has 3 default storages (free of charge)
  - \$VSC\_HOME
  - \$VSC\_DATA
  - \$VSC\_SCRATCH
- ✓ You can additionally request staging storage
- ✓ Different storage volumes have different:
  - mount point
  - size and performance
  - use case
  - backup and maintenance policy
- ✓ For more info, see the [ICTS Service Catalog](#) (EN/NL)

# Storage locations

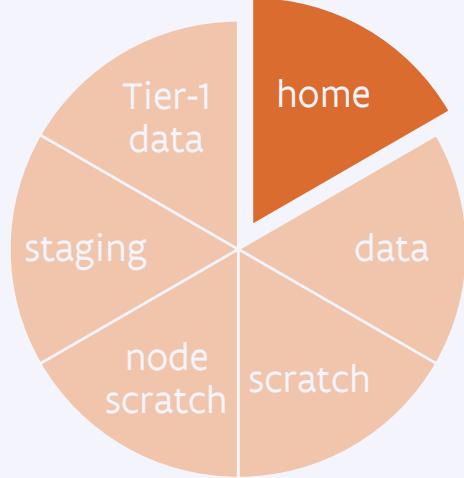
Example

- Avoid using `/tmp ($TMPDIR)` on the login nodes
  - It is only 10 GB and is needed by the operating system
  - Your application can crash if using `/tmp`
- You are automatically logged into your home folder upon login. Best to immediately switch to some other storage, e.g.  
`$ cd $VSC_DATA`
- Always check your storage balance using `myquota` command

```
$ myquota
file system $VSC_HOME
    Blocks: 1479M of 3072M
    Files: 12934 of 100k
file system $VSC_DATA
    Blocks: 12G of 75G
    Files: 1043k of 10000k
file system $VSC_SCRATCH
    Blocks: 15M of 1.5T
```

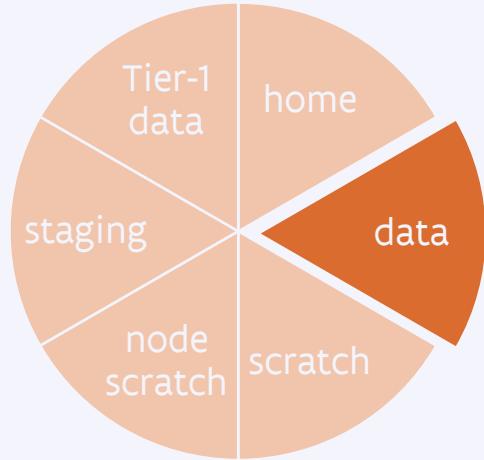
- [Request form for extra storage](#)
- [More info in the service catalog](#)

# Storage locations



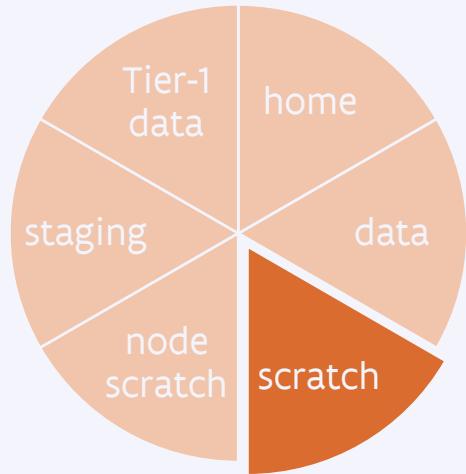
Storage	Home folder
Env. variable	\$VSC_HOME
Filesystem type	NFS
Access	Global
Backup	Hourly, daily, weekly (until last month) inside .snapshot folders
Default quotum	3 GB
Quotum increase	Not possible
Usage	Only storing SSH keys, config files, ...
Remarks	<ul style="list-style-type: none"><li>- Stay away from using it</li><li>- Can easily overflow:<ul style="list-style-type: none"><li>+ Your jobs may crash</li><li>+ Login issues</li></ul></li></ul>

# Storage locations



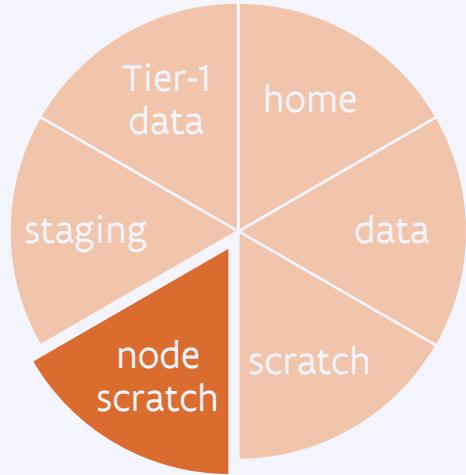
Storage	Data folder
Env. variable	\$VSC_DATA
Filesystem type	NFS
Access	Global
Backup	Hourly, daily, weekly (until last month) inside .snapshot folders
Default quotum	75 GB
Quotum increase	On purchase
Usage	Your data, code, software, results
Remarks	<ul style="list-style-type: none"><li>- Permanent storage for initial/final results</li><li>- Not optimal for intensive or parallel I/O</li></ul>

# Storage locations



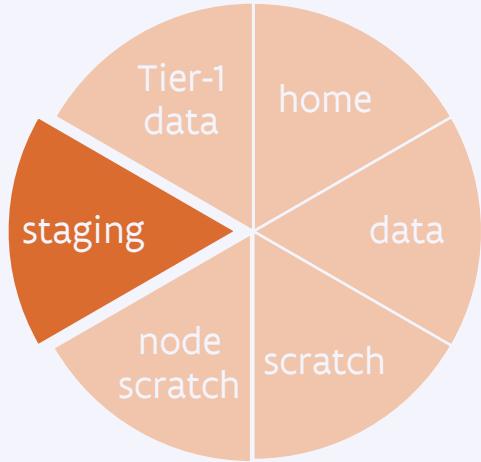
Storage	Scratch folder
Env. variable	\$VSC_SCRATCH
Filesystem type	Lustre
Access	Local
Backup	Files get deleted 30 days after last access
Default quotum	500 GB
Quotum increase	Free
Usage	Intensive, parallel I/O, temporary files
Remarks	<ul style="list-style-type: none"><li>- Recommended storage for all jobs</li><li>- Copy scratch files to VSC_DATA or local storage after jobs are done</li><li>- Deleted files cannot be recovered</li><li>- Avoid folders with &gt;10,000 files</li></ul>

# Storage locations



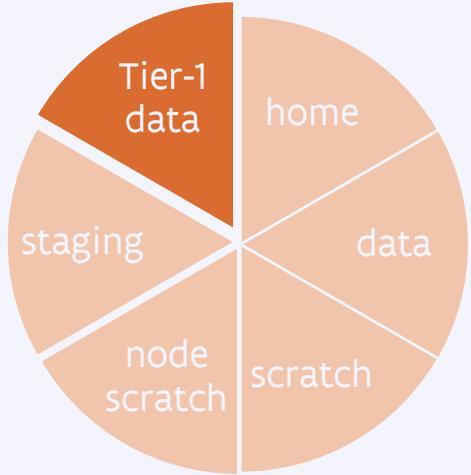
Storage	Node scratch folder
Env. variable	\$VSC_SCRATCH_NODE
Filesystem type	Ext4
Access	On compute node, only at runtime
Backup	None
Default quotum	591 GB
Extension	<a href="#">Read about beeOND</a>
Usage	Temporary storage at runtime
Remarks	<ul style="list-style-type: none"><li>- Fastest I/O, attached to the node</li><li>- Is cleaned after job terminates</li><li>- Copy the data to your home, scratch, or staging before job ends</li></ul>

# Storage locations



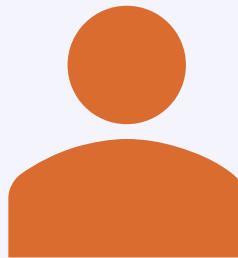
Storage	Staging folder
Path	/staging/leuven/stg_000xx
Filesystem type	Lustre
Access	On demand, only Tier-2@KUL
Backup	None
Default quotum	None
Extension	On purchase, from 1 TB
Usage	Permanent; share with a group
Remarks	<ul style="list-style-type: none"><li>- Accessible from login/compute nodes</li><li>- Fast, parallel I/O</li></ul>

# Storage locations



- VSC offers [Tier-1 Data](#) as a large storage service
- Only for active data (no archiving)
- Supports metadata (search & discover)
- Share within VSC (permission management)
- Share outside VSC (via [Globus](#))
- Stage-in/stage-out data workflow
- Free of charge for academic use
- Starting volume is 5 TB
- Submit a proposal to get access  
You can apply at any time

# Extra (KU Leuven) storage locations



Dataset



Archived package

- In production
- For **active** data
- Data should stage in/out
- Has backup
- See the [ICTS service catalog](#)

- In development
- For **inactive** data
- Storage until 10 years
- Has backup
- See the [ICTS service catalog](#)

Also see the [KU Leuven storage guide](#) for even more storage possibilities

# ManGO





## Compute credits

# Tier-2 credit system

What?

- Credits are a measure for compute resources
- Mandatory for accessing these resources
- Traceability and accountability

Why?

- Incentive to optimize your code and workflow

How?

- 1 million credits = 3.5 EUR
- Pay-as-you-go model
- Credit rates on the right = for fully allocated nodes  
(if you request less cores, you also get billed less)
- See the [ICTS service catalog](#) for more info

Node types	Credits per node-hour
Genius Cascadelake	10 000
Genius Skylake Bigmem	12 000
Genius Skylake 4 GPUs	20 000
Genius Skylake interactive node	Free
Genius Cascadelake 8 GPUs	39 900
Genius Superdome	126 000
wICE IceLake thin node	11 000
wICE IceLake Large Memory	19 000
wICE IceLake GPU node	45 000
wICE IceLake interactive node	Free
wICE IceLake hugemem node	80 000
wICE Sapphire Rapids	20 000
wICE AMD Zen-4 4 GPUs	150 000

# Tier-2 credit system – how to request credits

- Request your introduction credits
  - Every new user can request them once
  - Free of charge
  - 6 months

Check the [ICTS service catalog](#)

- Request permission to join a group

- Request project credits
  - Create group on the VSC account page
  - Start the name preferably with p\_-
  - Your group will start with lp\_-
  - Creator of the group can add and remove users

- Tier-2 credits can be used on all (Leuven) Tier-2 clusters

The image contains two screenshots of a web application interface for managing groups. Both screenshots feature a header with the Vlaams Supercomputer Centrum logo and navigation links for View Account, Edit Account, View Groups, New/Join Group, Edit Group, New/Join VO, Reservations, and Log Out.

**Screenshot 1: New/Join Group**

This screenshot shows the 'New/Join Group' page. A large orange arrow points from the text 'Check the ICTS service catalog' in the previous slide to this page. The page includes a note: '⚠ Your institute is Leuven / Hasselt, which means all the groups from your institute start with 'l'. Groups starting with any other letter are from other institutes. Join them at your own risk.' Below the note is a 'Create new group' button.

**Screenshot 2: Create new group**

This screenshot shows the 'Create new group' page. A red circle highlights the 'New/Join Group' button in the top navigation bar. The page has a note: '⚠ We will automatically prepend the letter 'l' to your groupname.' Below the note is a 'Groupname \*' input field.

# Tier-2 credit system – how

- Submit your jobs with a suitable credit account

```
#SBATCH -A lp_myproject
```

```
sbatch -A lp_myproject ... myjob.slurm
```

- Before submitting the job, Slurm checks if the account has enough available credits for the job  
(in addition to any other already-running jobs)
- Once a job starts running, it starts consuming credits from the account

# Tier-2 credit system – manage your credits

- We provide various sam-commands to facilitate credit related queries

Command	Purpose
sam-balance [-A account]	List active projects and available credits

```
tier2-p-login-3$ sam-balance -A lp_hpcinfo
ID          Name           Balance      Reserved      Available
===== ====== ====== ====== ======
96141      lp_hpcinfo    9782636        0            9782636
tier2-p-login-3$ 
```

# Tier-2 credit system – manage your credits

- We provide various sam-commands to facilitate credit related queries

Command	Purpose
sam-list-allocations -A <account-name>	List the credit deposits and refunds (if any)

```
tier2-p-login-3$ sam-list-allocations -A lp_hpcinfo
AllocID    AccountID Account          Timestamp           Credits
=====  ======  ======  ======  ======
80745      96141    lp_hpcinfo       2023-02-14T10:40:50  2000000
203565     96141    lp_hpcinfo       2023-04-17T16:42:26  2000000
227730     96141    lp_hpcinfo       2023-05-03T09:10:29  2000000
313474     96141    lp_hpcinfo       2023-06-19T17:33:59  8410000
tier2-p-login-3$ 
```

# Tier-2 credit system – manage your credits

- We provide various *sam*-commands to facilitate credit related queries

Command	Purpose
sam-quote sbatch [sbatch args] [myjob.slurm]	Estimates the (max) amount of credits a job will use

```
tier2-p-login-3$ sam-quote sbatch --cluster=wice hello.slurm  
10980  
tier2-p-login-3$ █
```

# Tier-2 credit system – manage your credits

- We provide various sam-commands to facilitate credit related queries

Command	Purpose
sam-statement -A <account-name> -s <start-date> -e <end-date>	Overview of credits used by each job in some time window

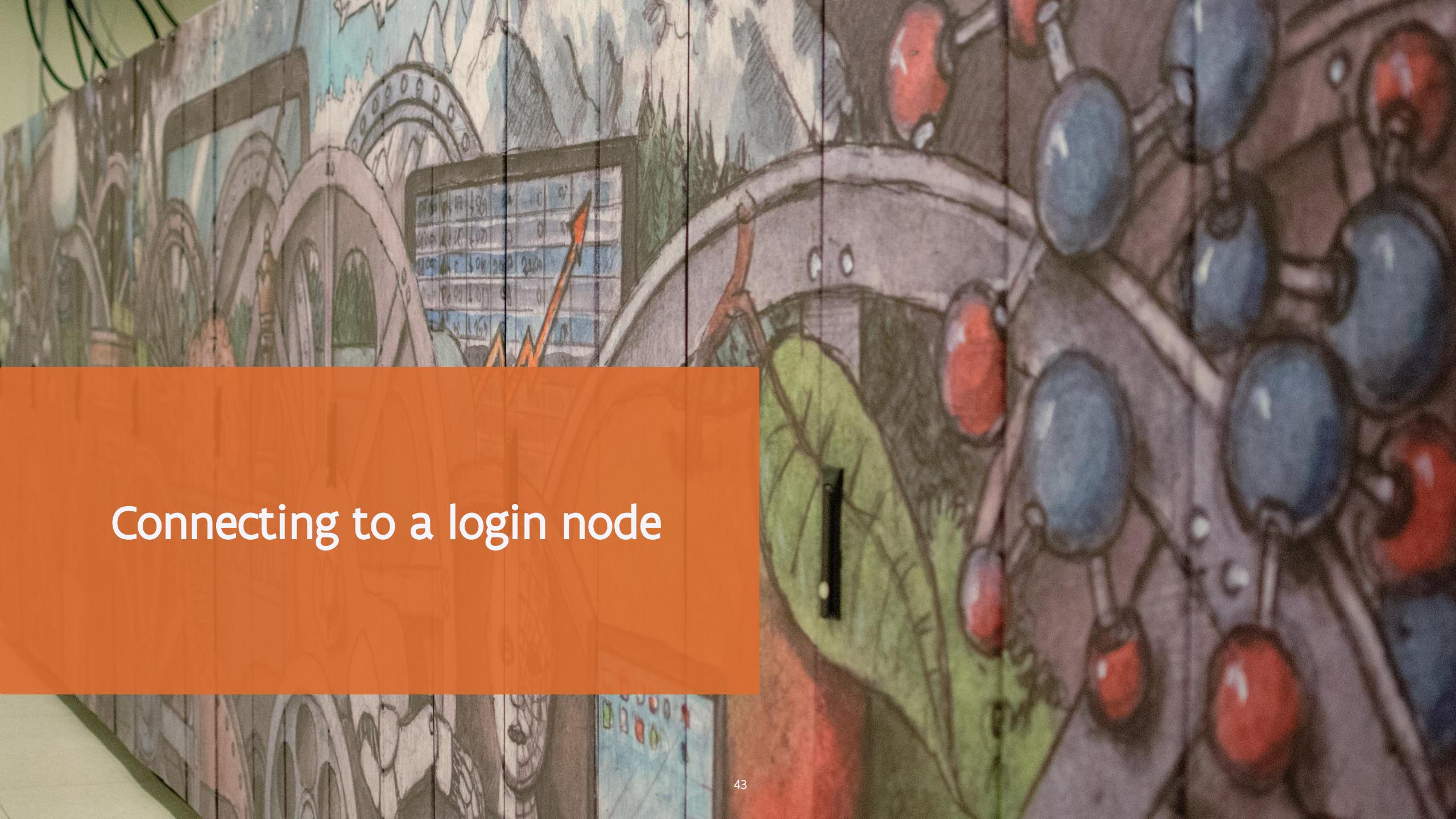
```
tier2-p-login-3$ sam-statement -A lp_hpcinfo -s 2023-09-01 -e 2023-10-25
#####
# Includes Account=lp_hpcinfo
# Generated on Mon Oct 30 11:01:29 2023.
# Reporting fund activity from 2023-09-01 to 2023-10-25
#
#####

Credits deposited in the given period: 0
Credits refunded in the given period: 0
Credits consumed in the given period: 0
-----
JobID   Cluster  Account        User      Partition  Credits
=====  ======  ======  ======  ======  ======
60739746  wice    lp_hpcinfo  vsc30446 batch    0
60740123  wice    lp_hpcinfo  vsc30446 batch    0
60740125  wice    lp_hpcinfo  vsc30446 batch    0
60746730  wice    lp_hpcinfo  vsc30446 interactive 0
60773547  wice    lp_hpcinfo  vsc30446 interactive 0
60774141  wice    lp_hpcinfo  vsc30446 batch    0
60774160  wice    lp_hpcinfo  vsc30446 batch    0
60783857  wice    lp_hpcinfo  vsc30446 interactive 0
```

# Tier-2 credit system – manage your credits

- We provide various sam-commands to facilitate credit related queries

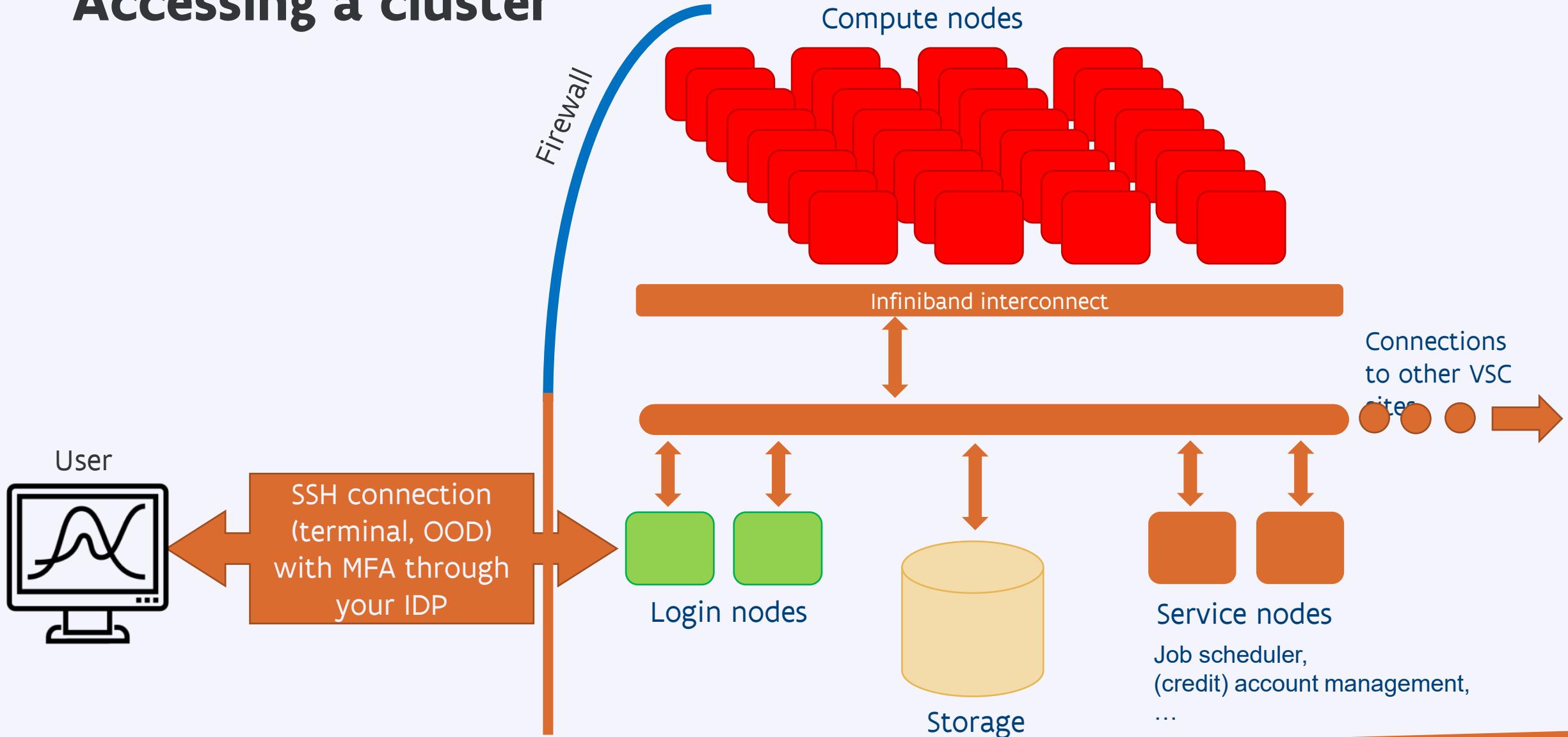
Command	Purpose
sam-balance	Lists active projects and available credits
sam-list-allocations -A <account-name>	Lists the credit deposits and refunds (if any)
sam-quote sbatch [sbatch args] [myjob.slurm]	Estimates the (max) number of credits a job will use
sam-statement -A <account-name> -s <start-date> -e <end-date>	Overview of credits used by each job in some time window



An orange rectangular overlay contains the text "Connecting to a login node".

Connecting to a login node

# Accessing a cluster



# How to login?

## Open OnDemand

- Web-based login via your browser
- <https://ondemand.hpc.kuleuven.be>
- Multi-Factor Authentication (MFA):  
Login through your institute's IDP
- No SSH key needed

## SSH client

- Windows: [PuTTY](#), [MobaXterm](#), ...  
MacOS/Linux: any terminal
- Optional: for VSC clusters *not* hosted at KU Leuven you need an [\(open\)SSH key pair](#) (upload the public key to the VSC account page and wait  $\geq$  30mins)
- Hosts: login.hpc.kuleuven.be:22  
Recommended to use an [SSH agent](#)
- Can open GUIs if X11 forwarding is enabled

# Using the login nodes

- To develop code
- To check your storage and credit balance
- To manage jobs (submit, check status, debug, resubmit, ...)
- To move data around
  - within VSC: use data, scratch, staging
  - outside VSC: copy/sync from/to your local storage (e.g. Globus)
- To pre- or postprocess your data
- To visualize your data
- To share files and folders

Tips

**Do NOT execute CPU- or memory-intensive tasks here**

Warning

Login nodes are shared resources

Submit jobs instead (you can e.g. compile software on compute nodes via an interactive job)

Use `slurmtop` to see how busy the cluster is

# Connecting via a terminal

(Linux and Mac)

- ✓ Use ssh to connect:

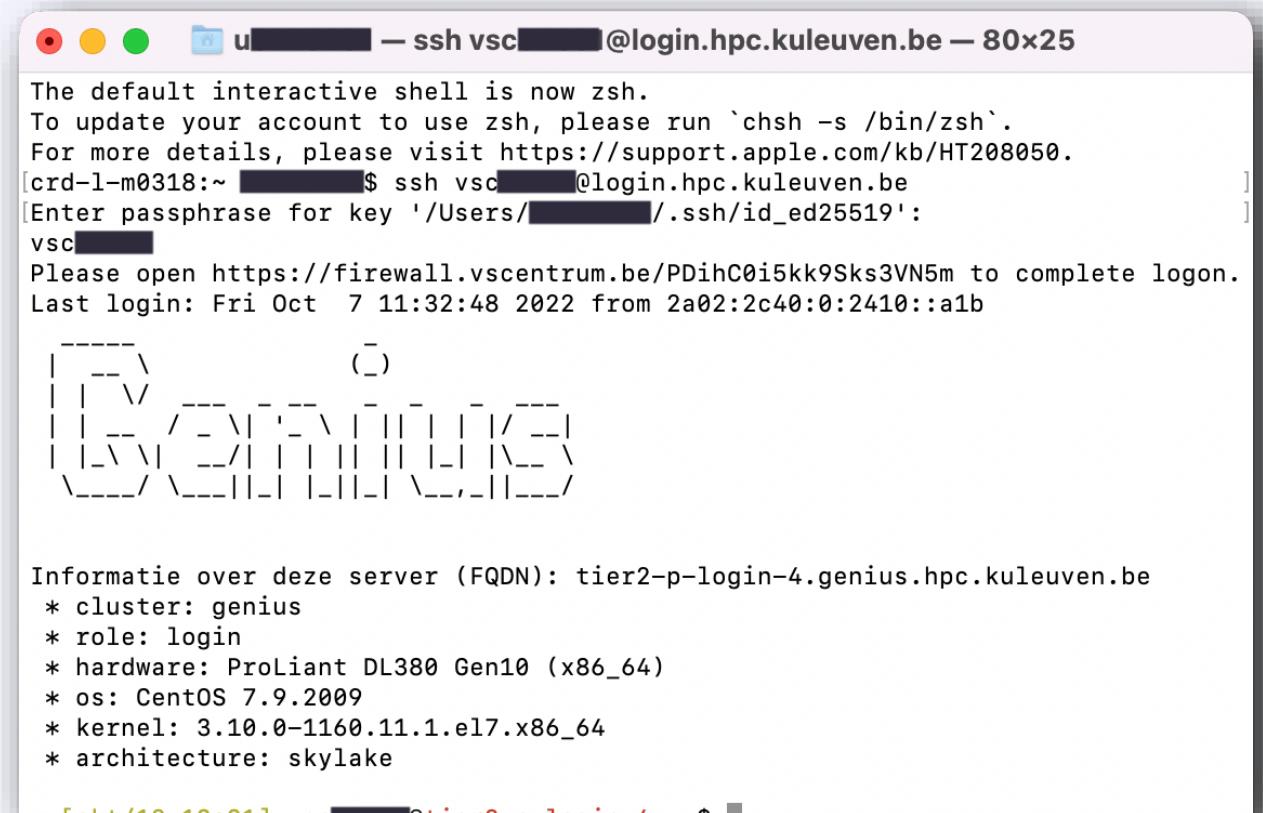
```
$ ssh -A vscXXXXX@<host_name>
```

- ✓ If key not found:

```
$ ssh -i </path/to/keyfile> ...
```

If asked for **password**, please stop connecting and contact support, otherwise after a few attempts you will be blocked for 24h.

**Hostname:**  
**login.hpc.kuleuven.be**



The default interactive shell is now zsh.  
To update your account to use zsh, please run `chsh -s /bin/zsh`.  
For more details, please visit <https://support.apple.com/kb/HT208050>.  
[crd-l-m0318:~ vscXXXXX\$ ssh vscXXXXX@login.hpc.kuleuven.be  
[Enter passphrase for key '/Users/vscXXXXX/.ssh/id\_ed25519':  
vscXXXXX  
Please open <https://firewall.vscentrum.be/PDihC0i5kk9Sks3VN5m> to complete logon.  
Last login: Fri Oct 7 11:32:48 2022 from 2a02:2c40:0:2410::a1b  
[REDACTED]  
Informatie over deze server (FQDN): tier2-p-login-4.genius.hpc.kuleuven.be  
\* cluster: genius  
\* role: login  
\* hardware: ProLiant DL380 Gen10 (x86\_64)  
\* os: CentOS 7.9.2009  
\* kernel: 3.10.0-1160.11.1.el7.x86\_64  
\* architecture: skylake  
✓ [okt/12 13:31] vscXXXXX@tier2-p-login-4 ~ \$

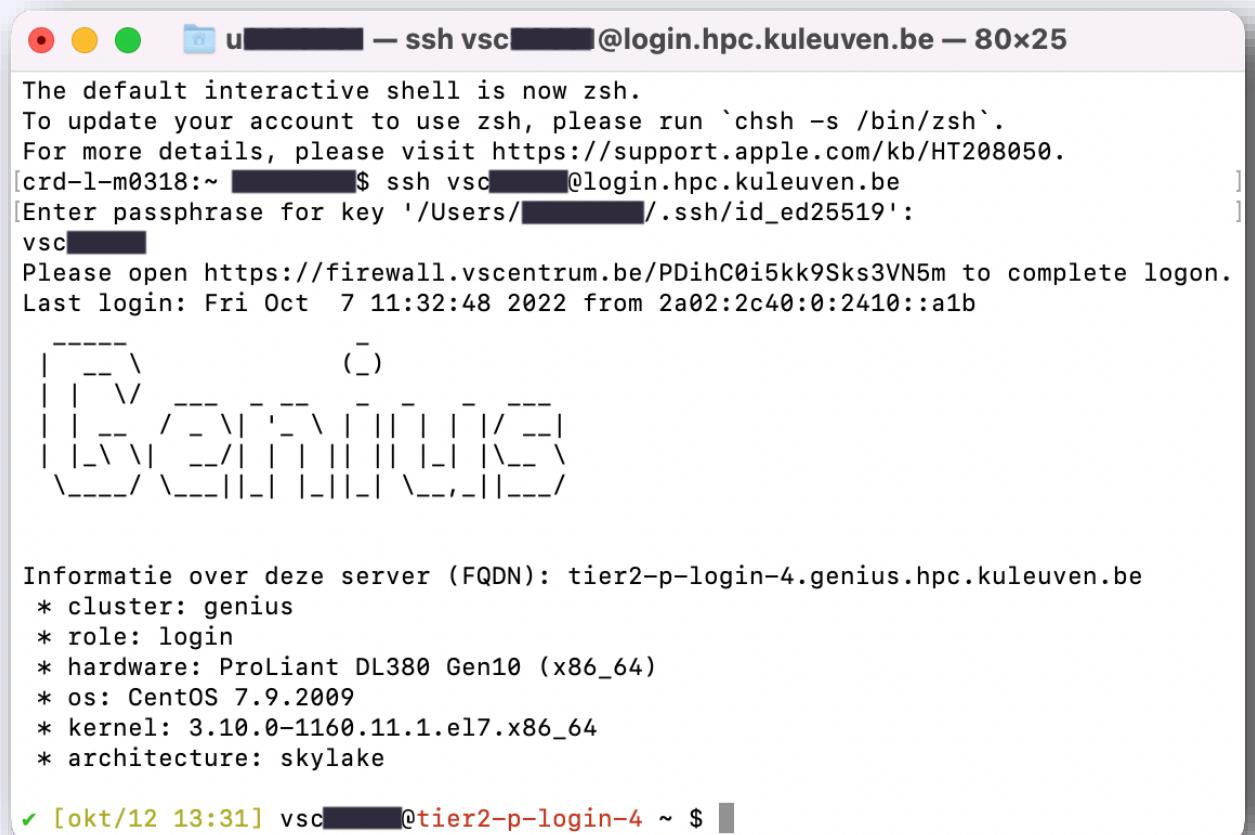
# Connecting via a terminal

(Linux and Mac)

**Hostname:**  
**login.hpc.kuleuven.be**

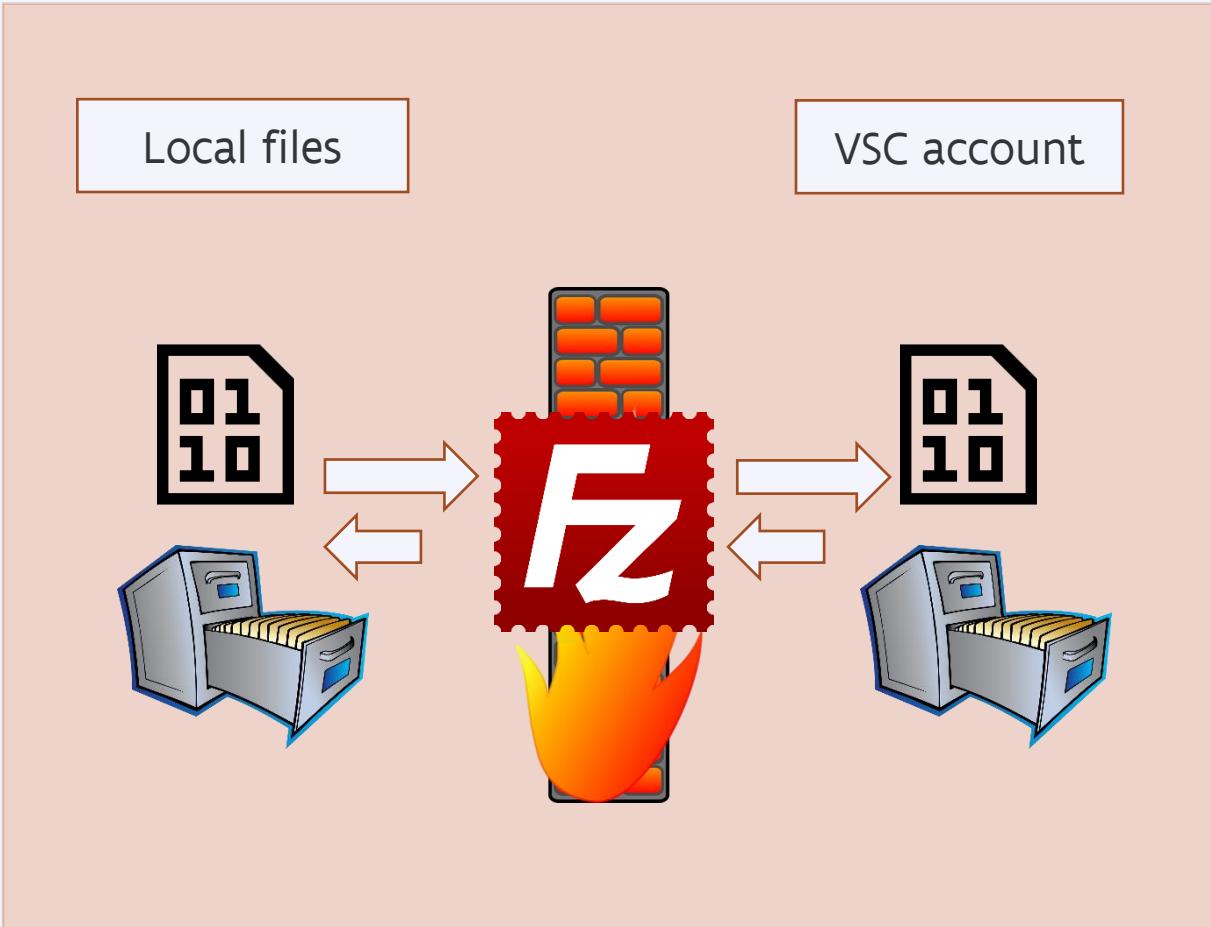
# With an SSH agent

- ✓ Check your SSH agent  
Is your SSH key found?  
\$ ssh-add -l
  - ✓ If your SSH agent is not running:  
\$ eval \$(ssh-agent)
  - ✓ If your key is not found,  
add it to the agent:  
\$ ssh-add </path/to/keyfile>
  - ✓ Use ssh to connect:  
\$ ssh -A vscXXXXX@<hostname>



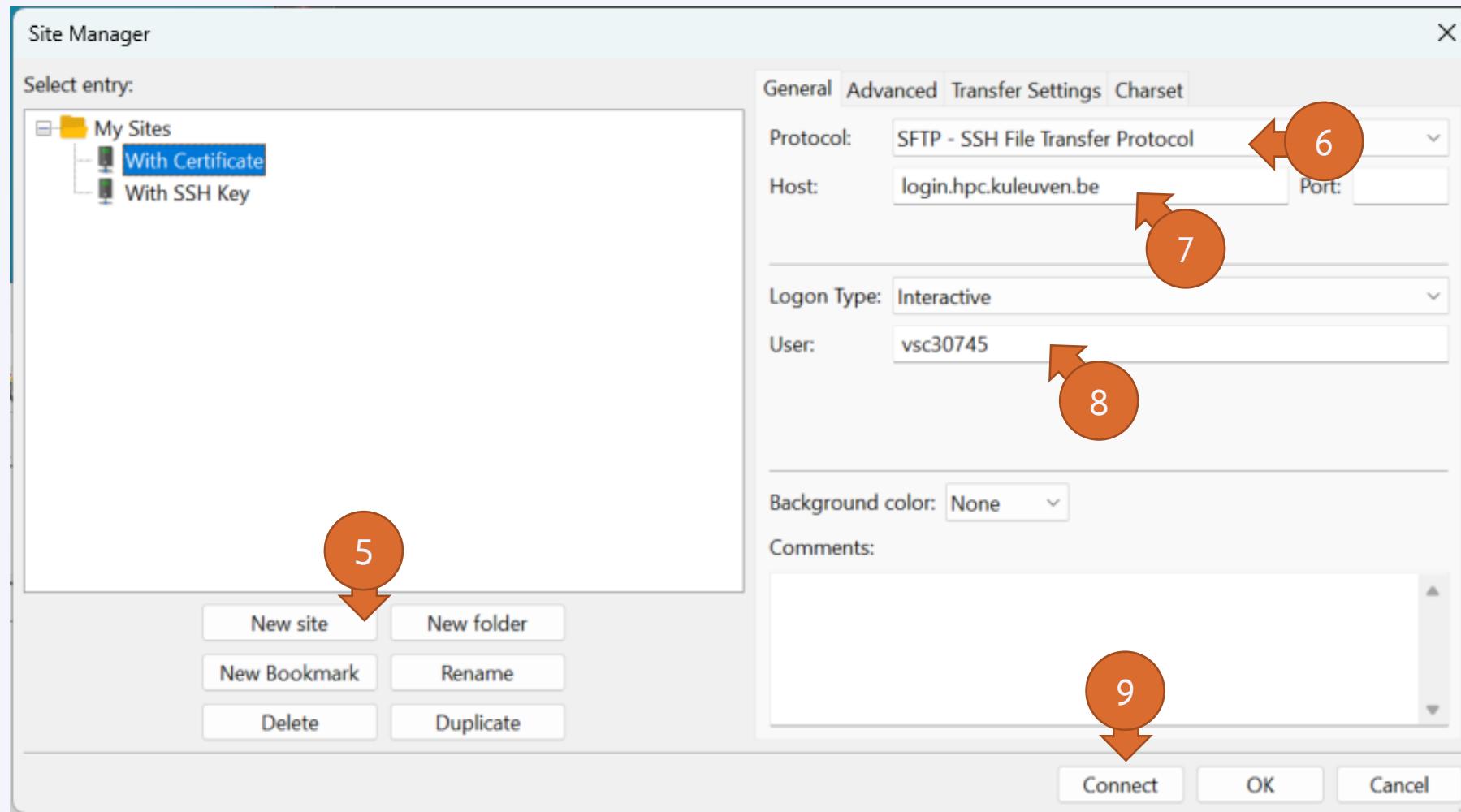
# Data transfer

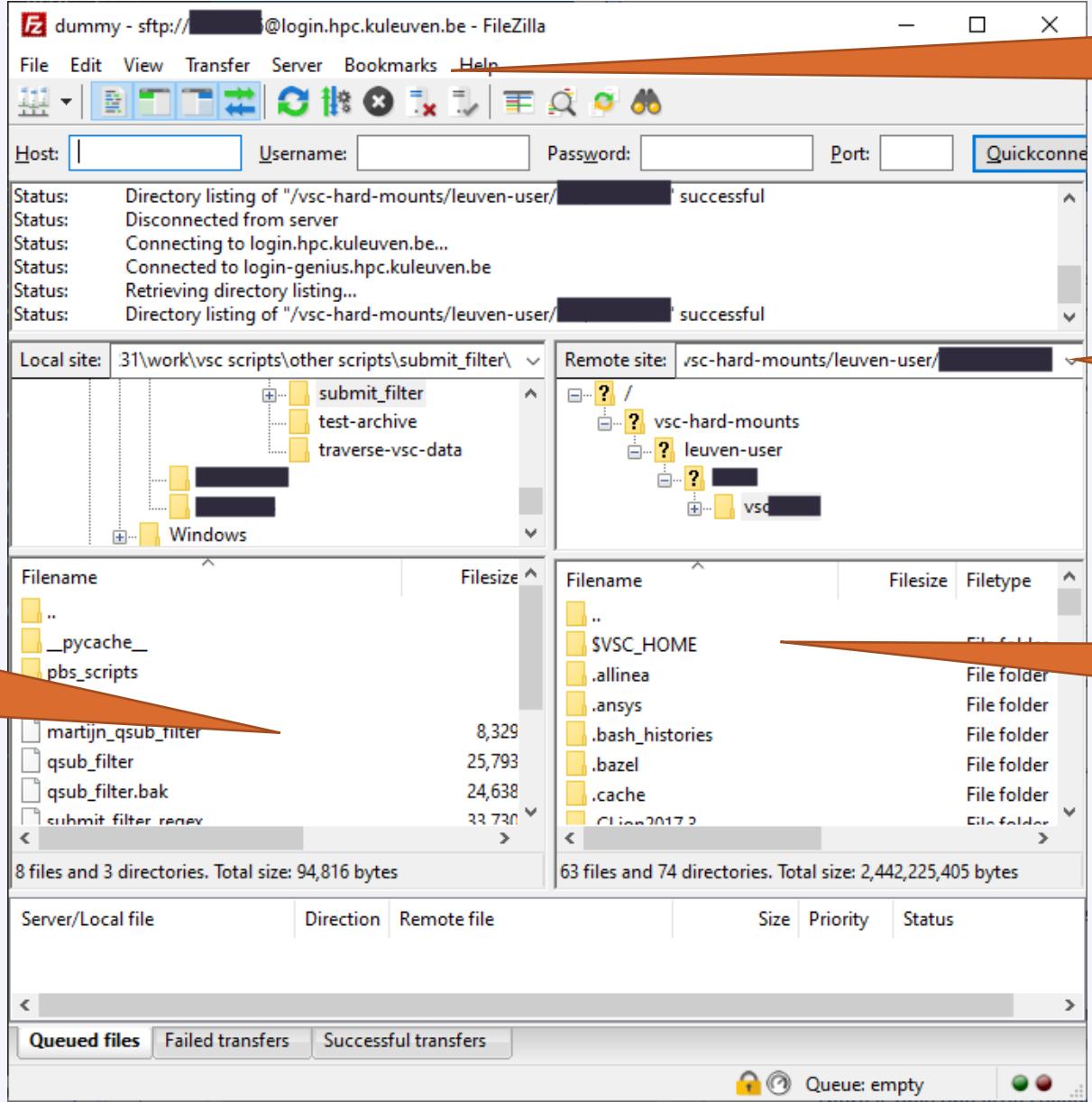
# File transfer tools



Application	OS
<a href="#">FileZilla</a>	Windows, Linux, Mac
<a href="#">WinSCP</a>	Windows
rsync, scp	Linux, Mac
<a href="#">Globus</a>	Window, Linux, Mac

# File transfer with FileZilla





Content on your local machine (e.g. laptop)

It can be convenient to bookmark your data and scratch folders

Navigate here to to your \$VSC\_DATA folder, e.g. /data/leuven/399/vsc39934

Content of your chosen folder

**WARNING:** by default you will land in your \$VSC\_HOME which is typically NOT where you want to store data

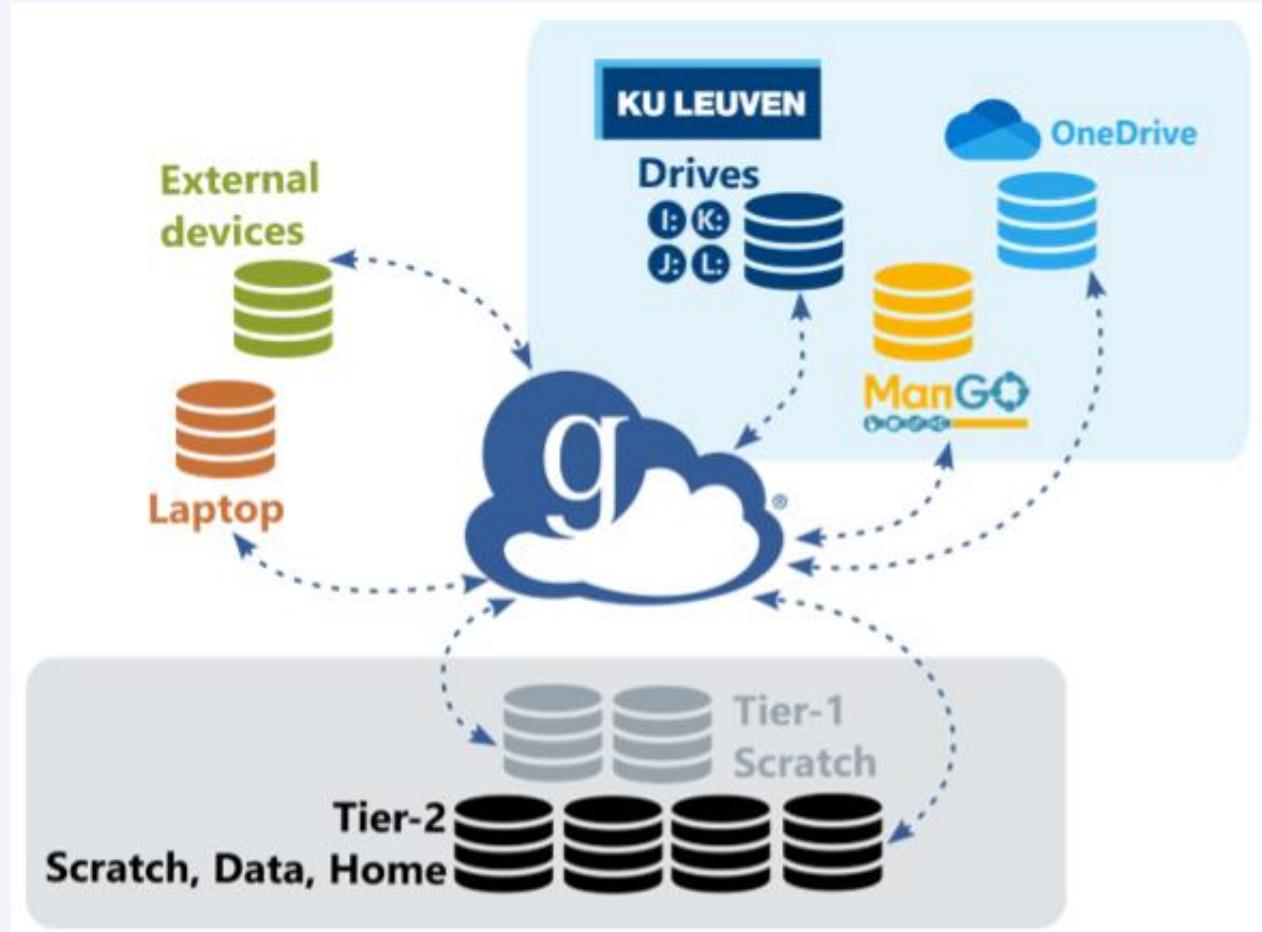
# File transfer with rsync

- ❑ For Linux, Mac and Windows WSL users
- ❑ Synchronize source and destination
- ❑ Listens to your SSH agent
- ❑ Example usage:

```
$ rsync -av /path/to/source vsc3XXXX@login.hpc.kuleuven.be:/path/to/destination
```

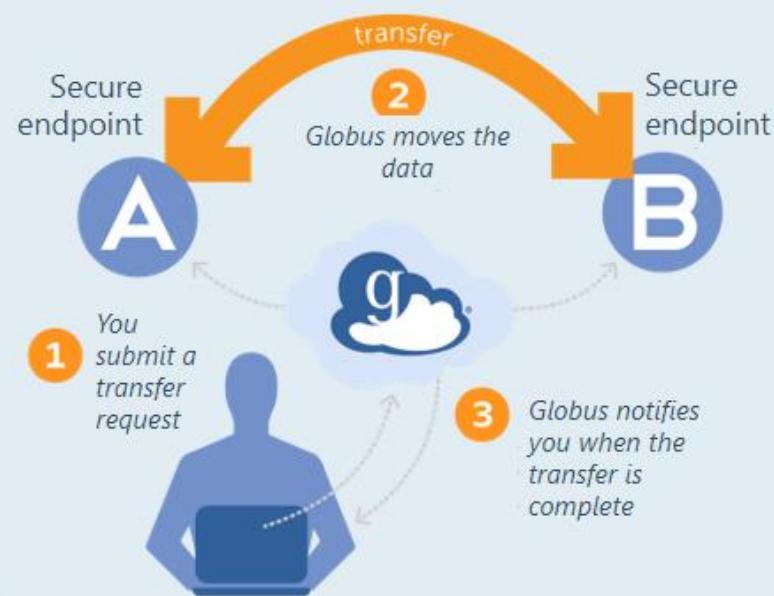
# File transfer with Globus

- Web interface for transferring, sharing and searching within data
  - Schedule and resume (large) transfers
  - Workflow:
    - Define “endpoint” on the source
    - Define “endpoint” on the destination
    - Transfer between endpoints
  - Endpoints on all VSC sites:
    - all Tier-2 clusters: home, data, scratch\*
    - all Tier-1 clusters: scratch, projects
- \* You can also access staging storage through this endpoint (Leuven Tier-2 only)

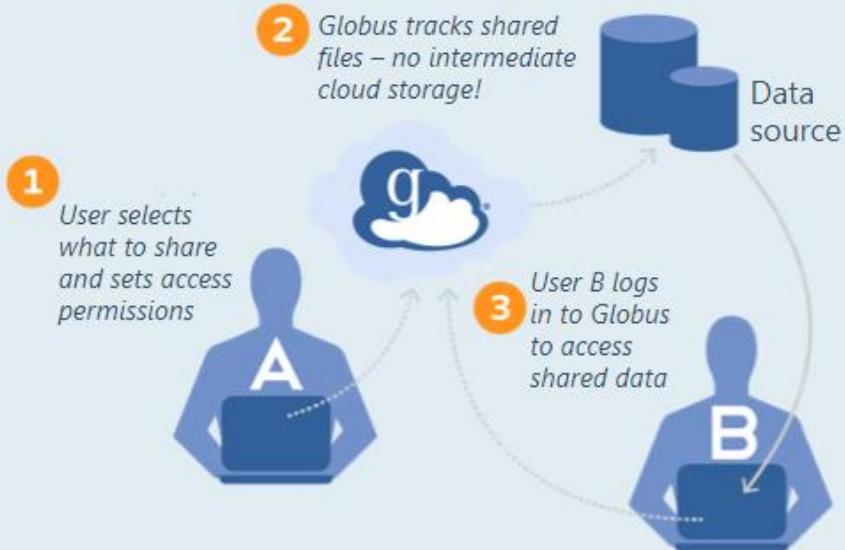




## Transfer data



## Share data





An orange rectangular overlay is positioned in the lower-left quadrant of the image, containing the text "Open OnDemand".

Open OnDemand

# Open OnDemand

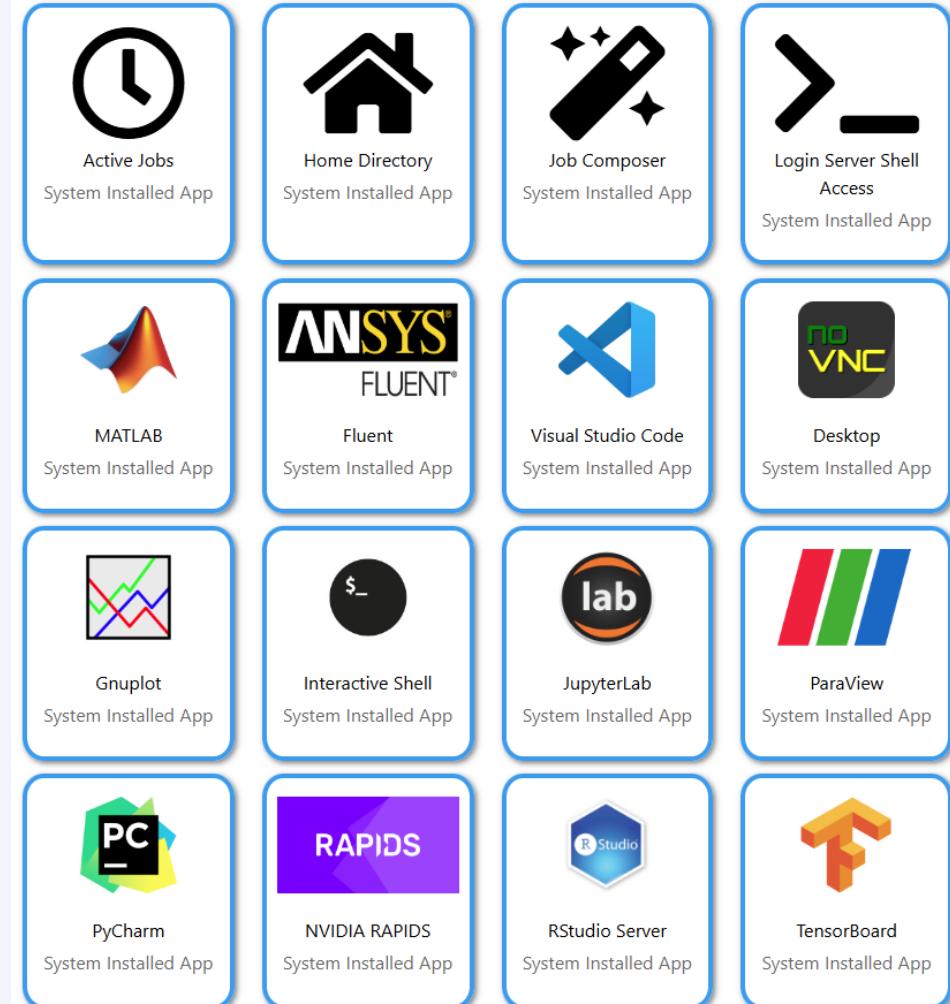
- ✓ Access clusters via your web browser
- ✓ <https://ondemand.hpc.kuleuven.be>
- ✓ Login via your institute's IDP
- ✓ File browser + Data Transfer (Globus)
- ✓ Develop, compile and test your code
- ✓ Interactive apps + integrated shell
- ✓ noVNC Desktop for graphical apps
- ✓ VSCode and PyCharm editors
- ✓ Fluent, Jupyter Lab, Rstudio, MATLAB, ParaView, Tensorboard, Gnuplot

Please close your interactive sessions  
when you no longer actively use them!

KU LEUVEN

Open OnDemand provides an integrated, single access point for all your HPC resources.

Pinned Apps A featured subset of all available apps





# Open OnDemand

Access your VSC storage

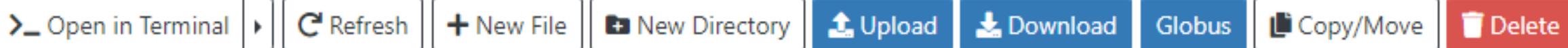
- ✓ Use the “Files” tab to access your VSC\_HOME and VSC\_DATA
- ✓ VSC\_SCRATCH is not available from this menu

KU Leuven OnDemand Apps ▾ Files ▾ Jobs ▾ Clusters ▾ Interactive Apps ▾

Home Directory  
Data Directory /data/leuven/307/vsc30745

KU LEUVEN

File/Folder management



- ✓ OK for transferring small files/folders
- ✓ Open any (sub)folder in terminal (will open a new tab)
- ✓ Deleted files/folders can be retrieved from (hourly, daily, weekly) snapshots



Monitor your active jobs

KU Leuven OnDemand

Apps ▾

Files ▾

Jobs ▾

Clusters ▾

Interactive Apps ▾



⌚ Active Jobs

📝 Job Composer

Your Jobs ▾

All Clusters ▾

## Active Jobs

Show  entries

Filter:

ID	Name	User	Account	Time Used	Queue	Status	Cluster	Actions
----	------	------	---------	-----------	-------	--------	---------	---------



55337242

PostProcessing

vsc30745

lp\_hpcinfo

00:02:03

batch

Completed

genius



# Open OnDemand: Login Server Shell Access

- ✓ Starts a shell in a new browser tab
- ✓ You land on a Genius login node  
(from which you can also access wICE)
- ✓ You land on your VSC\_HOME  
cd to data/scratch/staging  
immediately
- ✓ Do NOT compute on the login nodes
- ✓ Try:  
\$ sam-balance  
\$ myquota

Login shell

Host: login.hpc.kuleuven.be      Themes: Default

```
[vsc30745@login: ~]$
```

Informatie over deze server (FQDN): tier2-p-login-1.genius.hpc.kuleuven.be

- \* cluster: genius
- \* role: login
- \* hardware: ProLiant DL380 Gen10 (x86\_64)
- \* os: Rocky 8.8
- \* kernel: 4.18.0-477.10.1.el8\_8.x86\_64
- \* architecture:
- \* architecture-suffix:

On 25-09-2023 the operating system of the Genius cluster was upgraded to Rocky8.  
The following documentation page provides an overview of important changes:  
[https://docs.vsczentrum.be/leuven/genius\\_2\\_rocky.html](https://docs.vsczentrum.be/leuven/genius_2_rocky.html)  
Last login: Tue Nov 7 23:42:32 2023 from 2a02:2c40:0:2410::a1c

```
vsc30745@tier2-p-login-1 ~
```

```
$ [red]
```



# Open OnDemand: Interactive Shell

- ✓ Start a shell as a job (on a compute node)
- ✓ Recommended to:
  - Install your own software (in VSC\_DATA)
  - Test/run your program interactively
  - Debug a program
- ✓ Choose the relevant cluster, partition and resources (cores, memory, GPU)

## Interactive Shell

Interactive Apps

Servers
Interactive Shell
Jupyter Lab
Nvidia Rapids
RStudio Server
Tensorboard
code-server

Cluster: wice

Account: lpt2\_sysadmin

Partition: interactive

Number of hours: 1

Number of cores: 1

Required memory per core in megabytes: 3400

Number of nodes: 1

Number of gpu's: 0

[type:] Specify the total number of GPUs slices for the job. An optional GPU type specification can be supplied. For example "A100:3" or "3".

Reservation (optional):

Name of an existing reservation in which the job should run

I would like to receive an email when the session starts

**Launch**

\* The Interactive Shell session data for this session can be accessed under the [data root directory](#).



# Open OnDemand: JupyterLab

## Jupyter Lab

- ✓ [Create a notebook](#) as a job (on a compute node)
- ✓ Suitable for:
  - pre/post-processing your data
  - prototyping
  - adding figures and text for pedagogical purposes
- ✓ Choose the appropriate cluster, partition and resources (cores, memory, GPU)
- ✓ User your own Python/R kernels (next slide)

Interactive Apps

Servers

- Interactive Shell
- Jupyter Lab**
- Nvidia Rapids
- ▲ RStudio Server
- ◆ Tensorboard
- ◀ code-server

Cluster

genius

Account

lpt2\_sysadmin

Partition

gpu\_p100

batch(\_long) or bigmem or interactive or gpu(\_p100|\_v100) or dedicated...

Enable nvidia rapids

Only available on w/CE interactive or gpu nodes with a GPU requested

Number of hours

1

Number of cores

9

Required memory per core in megabytes

3400

Number of nodes

1

Number of gpu's

1

[type:] Specify the total number of GPUs slices for the job. An optional GPU type specification can be supplied. For example "A100:3" or "3".

Reservation (optional)

Name of an existing reservation in which the job should run

I would like to receive an email when the session starts

Pre-run scriptlet

Bash commands to be executed before starting application, \$node,\_arch variable containing for example 'skylake' is available

**Launch**

\* The Jupyter Lab session data for this session can be accessed under the [data root directory](#).



# Open OnDemand: Custom R/Python kernels

Custom kernels

- ✓ The default kernels may not offer the R/Python/... interpreter version or packages you want
- ✓ You can create a miniconda env and add it to your OnDemand kernels

Step 1: Miniconda

Start an Interactive Shell

Go to your data folder

```
$ cd ${VSC_DATA}
```

Download Miniconda

```
$ wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86\_64.sh
```

Install Miniconda in your VSC\_DATA

```
$ bash Miniconda3-latest-Linux-x86_64.sh -b -p ${VSC_DATA}/miniconda3
```

Permanently add the path to Miniconda to your ~/.bashrc

```
$ echo 'export PATH="${VSC_DATA}/miniconda3/bin:${PATH}"' >> ~/.bashrc
```



# Open OnDemand: Custom R/Python Kernels

## Step 2: Environment setup

Create a new environment <env\_name> with all packages you need, e.g.

```
$ conda create --name=<env_name> numpy scipy ...
```

Enable your new environment

```
$ source activate <env_name>
```

Add the ipykernel package

```
$ conda install -c conda-forge ipykernel
```

## Step 3: Register your kernel

```
$ python -m ipykernel install --user --env PYTHONPATH "" \  
--name <env_name> --display-name <env_name>
```

## Step 4: Launching JupyterLab

Now start a new Jupyter Lab session

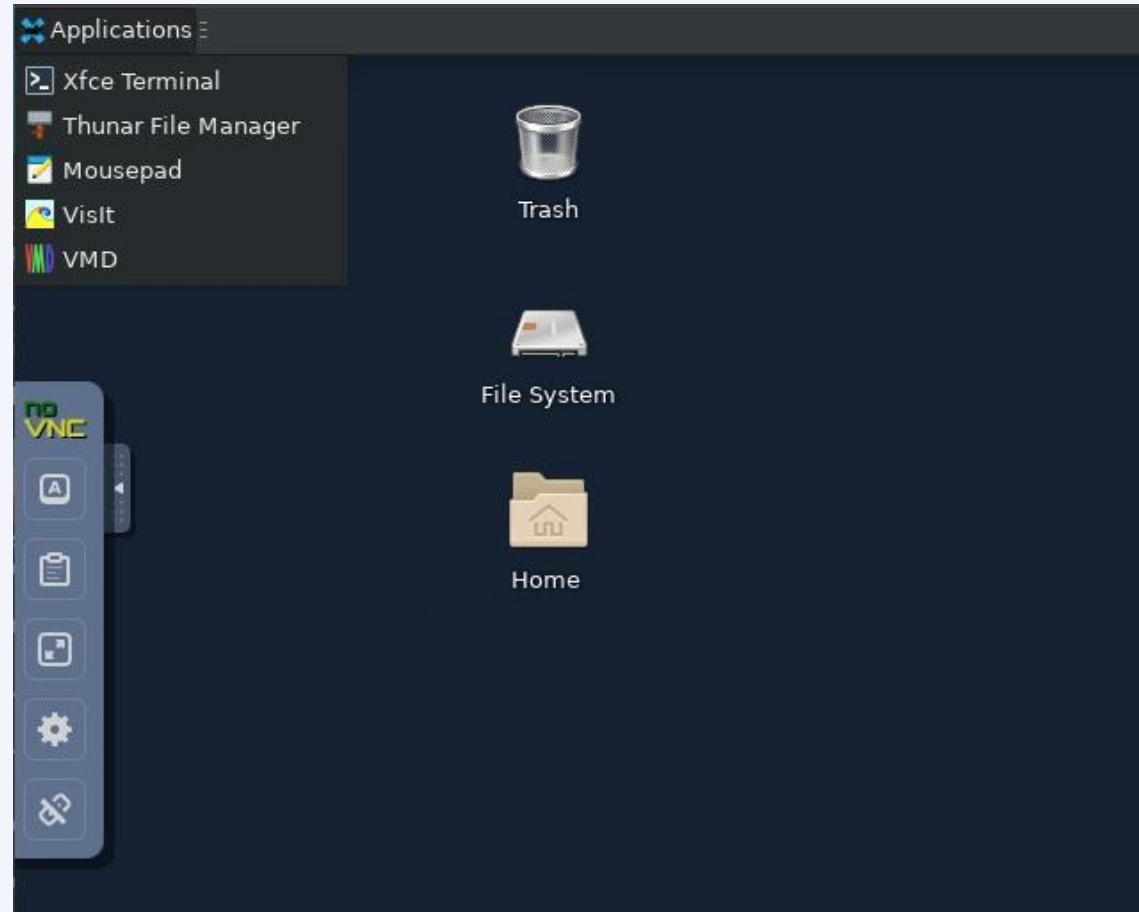
Your new kernel should be available there



# Open OnDemand: noVNC Desktop

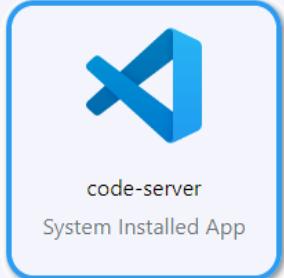
Desktop

- ✓ Ideal for apps with GUI
- ✓ Replacement for NoMachine/NX
- ✓ Minimalistic Terminal (different from login node)
- ✓ Load (graphical) modules in Terminal, e.g. MATLAB, Fluent, ncview, VisIt, ...
- ✓ Mousepad: basic file editor

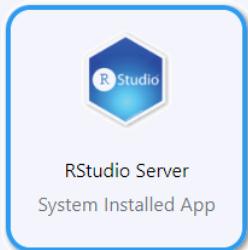




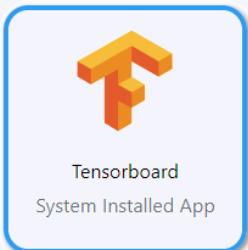
# Open OnDemand: Other apps



Start Visual Studio Code server as a job  
Develop, deploy and debug your workflow directly on HPC  
Use of the interactive partitions is free of charge and is a good choice  
Supports many languages + Github integration



Start RStudio IDE as a job  
Install your own packages in VSC\_DATA



Tensorboard  
System Installed App

Track machine learning metrics and visualize data during the workflow  
You need to provide a log folder



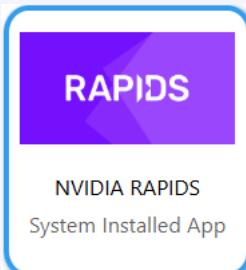
# Open OnDemand: Other apps



Start PyCharm IDE as a job  
Develop, deploy and debug your Python workflow directly on HPC  
Use Python environment based on Conda or virtual environments



Use Fluent for CFD simulations (single-node)  
You need to be in a license group before being able to see the app logo  
Contact your supervisor to help you joining an existing license group



Use RAPIDS for accelerated data types and functionalities from NumPy, Pandas (and more) on GPUs



# Open OnDemand: Resources

E.g. to start a Jupyter notebook

- ✓ Pick a valid Slurm credit account
- ✓ Default partition for most apps: *interactive*  
(interactive: max 8 cores, 1 GPU instance, 16 hr)
- ✓ Default resources:  
1 core, 1 hour, 3400 MB RAM, no GPU

Interactive Apps

Servers
● Interactive Shell
<b>Jupyter Lab</b>
● RStudio Server
● Tensorboard
● code-server
Work in progress
● ParaViewWeb - Work in progress
● cryosparc

## Jupyter Lab

This app will launch a Jupyter Lab server on one or more nodes.

### Account

### Partition

batch(\_long) or bigmem or interactive or gpu or dedicated...

### Number of hours

### Number of cores

### Required memory per core in megabytes

### Number of nodes

### Number of gpu's

[type:] Specify the total number of GPUs slices for the job. An optional GPU type specification can be supplied. For example "A100:3" or "3".

### Reservation (optional)

Name of an existing reservation in which the job should run

I would like to receive an email when the session starts

# Software

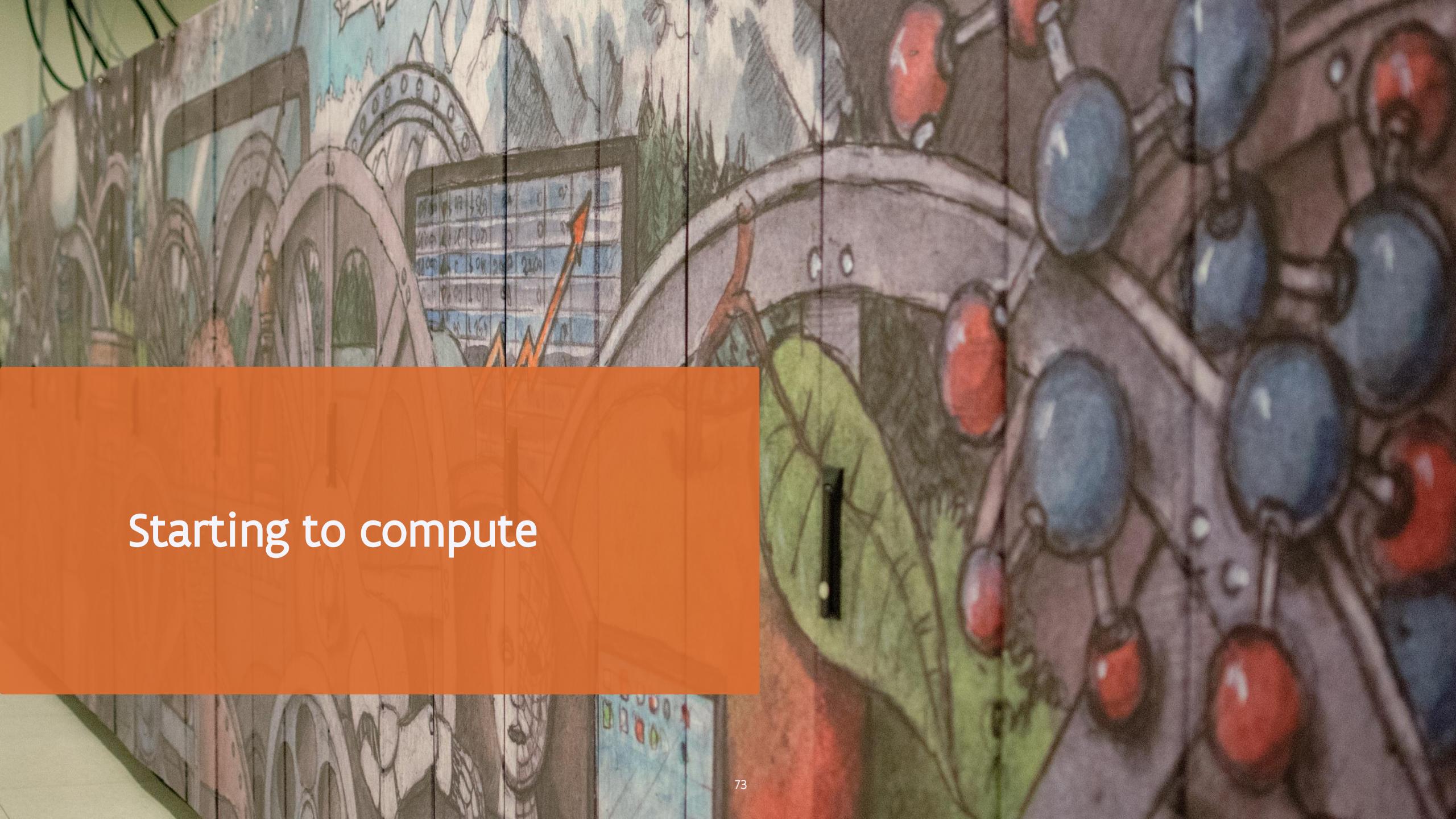
# Software: Centrally installed modules

- ✓ OS: Rocky Linux 8.10
- ✓ Toolchains:
  - *intel* (icc, icpc, ifort; Intel MPI; Intel MKL)
  - *foss* (gcc, g++, gfortran; OpenMPI; ScaLAPACK, OpenBLAS, FFTW)
  - Toolchain year on Genius from 2018a onwards; on wlCE from 2021a onwards
- ✓ Note: not all toolchains are compatible with each other (conflicting dependencies)  
For example, **never mix ...-*foss*... and ...-*intel*...** modules

Command	Remark
module av	List all available modules
module av Python	List all Python-related modules
module spider Python	Get more info
module load Python/3.12.3-GCCcore-13.3.0	Load a specific module
module list	List all loaded modules and their dependencies
module unload Python/3.12.3-GCCcore-13.3.0	Unload a module (but dependencies still stay)
module purge	Remove all modules from your work session

# Software: Your specific needs

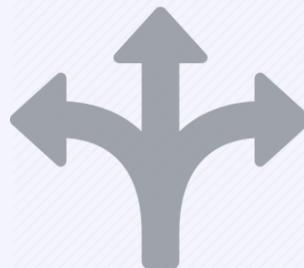
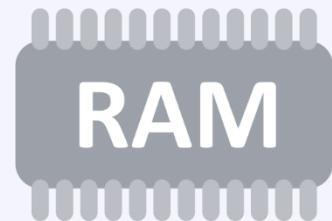
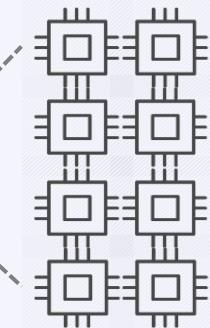
- ✓ If you need a software package that is not yet available as a module, you can ask us to provide it
  - ✓ For Python and R packages, however, we prefer that you install it yourself
  - ✓ Read more about [Python Package Management](#)
  - ✓ Read more about [R Package Management](#)
- ✓ If you want to compile software yourself:
  - ✓ Best install it in your VSC\_DATA
  - ✓ Use *intel* or *foss* toolchains
  - ✓ Compile your code on a compute node (with interactive job)
  - ✓ If you run into problems, don't hesitate to contact us



Starting to compute

# Resource glossary

- ✓ **Cluster:** which machine to use? Genius or wICE?
- ✓ **Nodes:** how many compute servers to allocate?
- ✓ **Cores:** how many cores per node to allocate?
- ✓ **Memory:** how much CPU memory to allocate?
- ✓ **Partition:** batch, gpu, bigmem, ...?
- ✓ **Walltime:** how long to allocate the resources?
- ✓ **Storage:** how much storage (data, scratch, etc) will the job require?
- ✓ **Credits:** how many compute credits will the job consume?

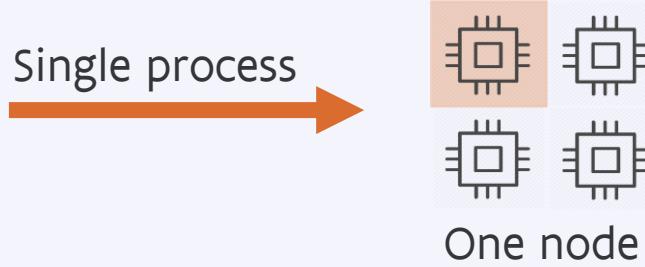


# Default resources

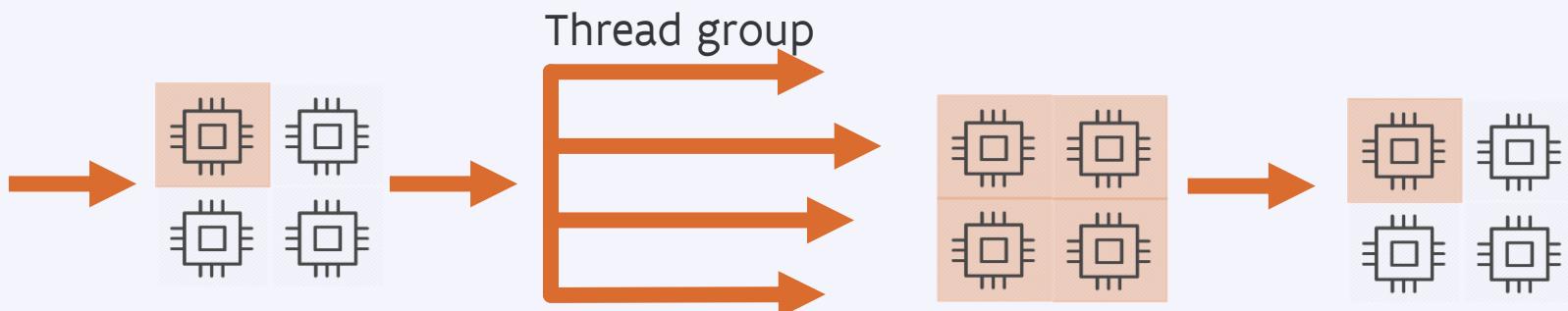
- ✓ Cluster: the cluster associated with the node you're on (Genius or wICE)
- ✓ Nodes: 1
- ✓ Cores: 1
- ✓ Memory: depends on which node(s) you use
- ✓ Partition: batch
- ✓ Walltime: typically 1 hour
- ✓ Storage: no default
- ✓ Credits: no default



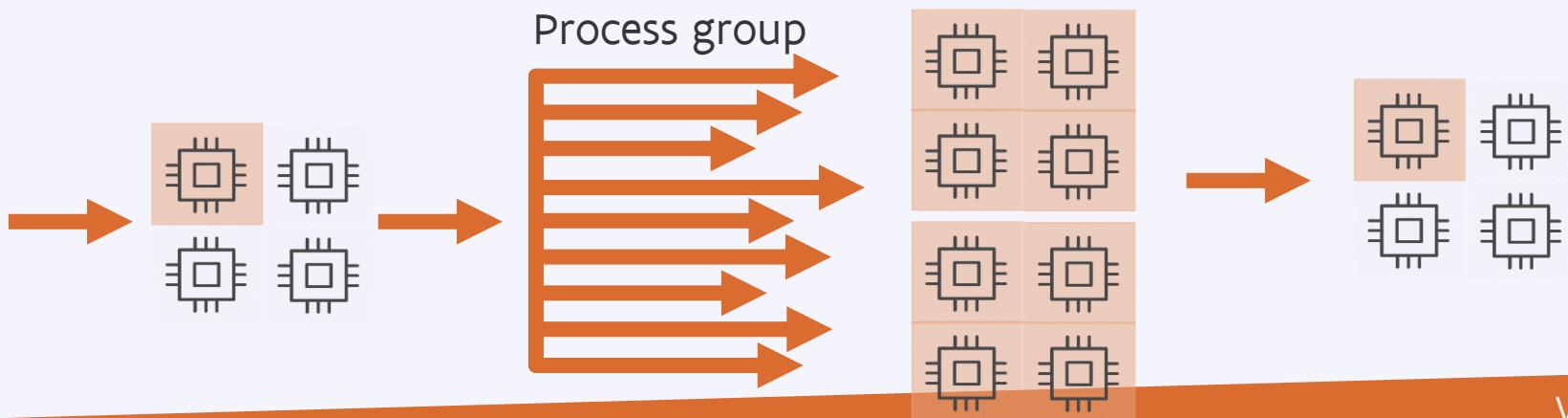
**Serial application**  
1 process on 1 core



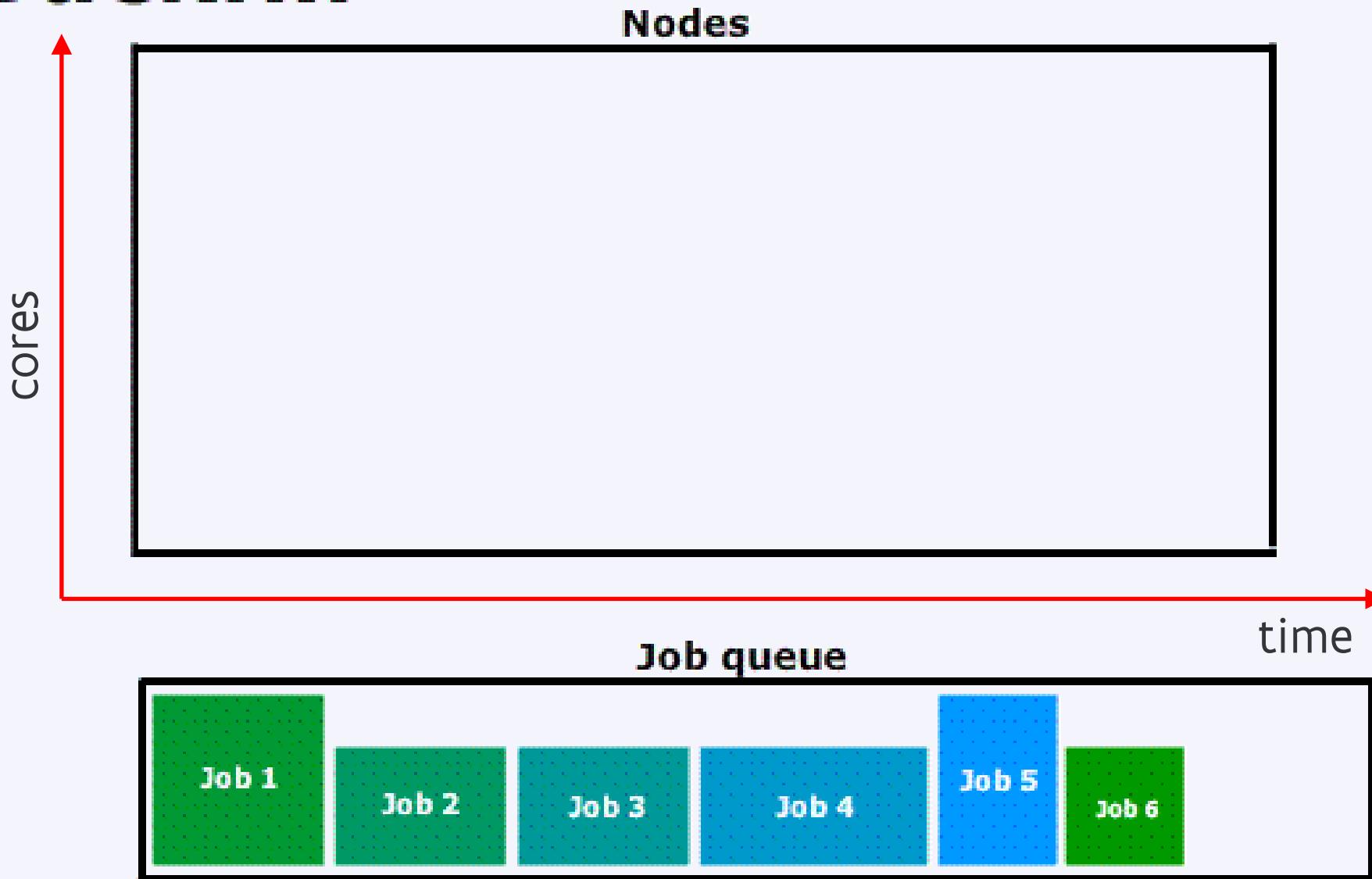
**Multi-threaded application**  
N threads on N cores  
on 1 node



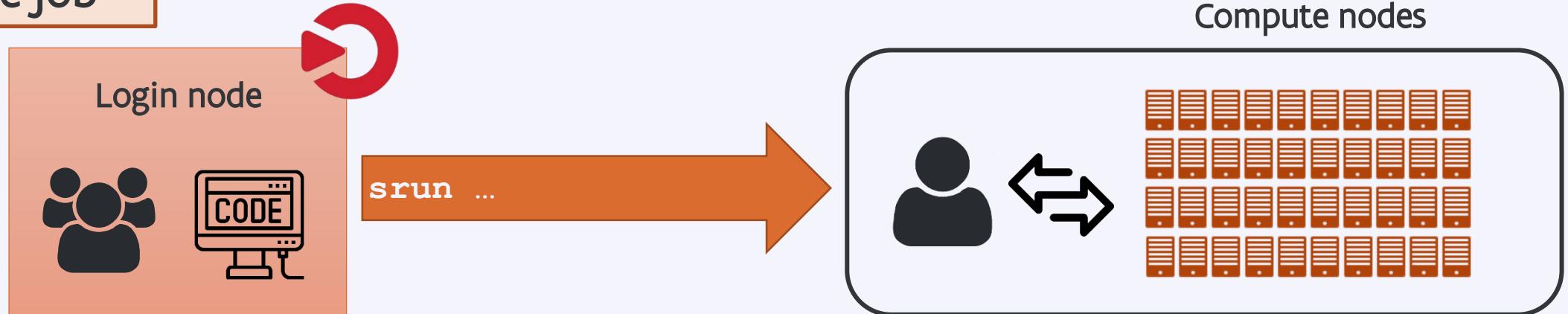
**Distributed application**  
Many processes on  
multiple cores,  
possibly spread over  
multiple nodes



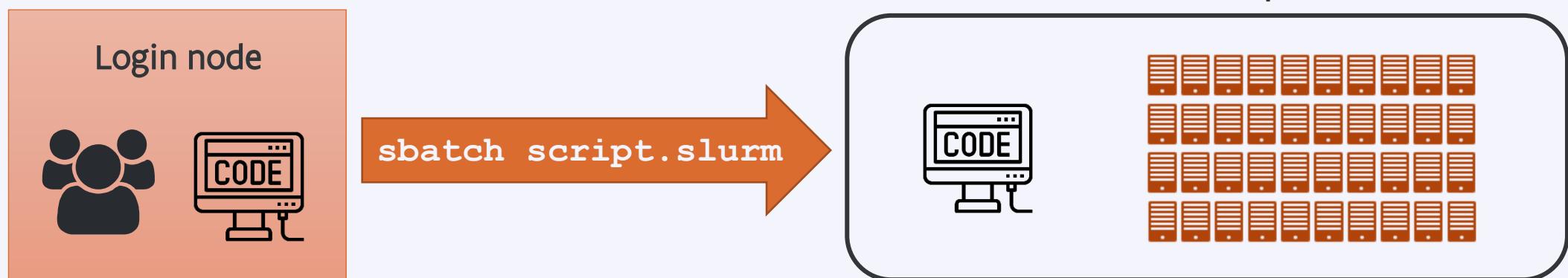
# Backfill



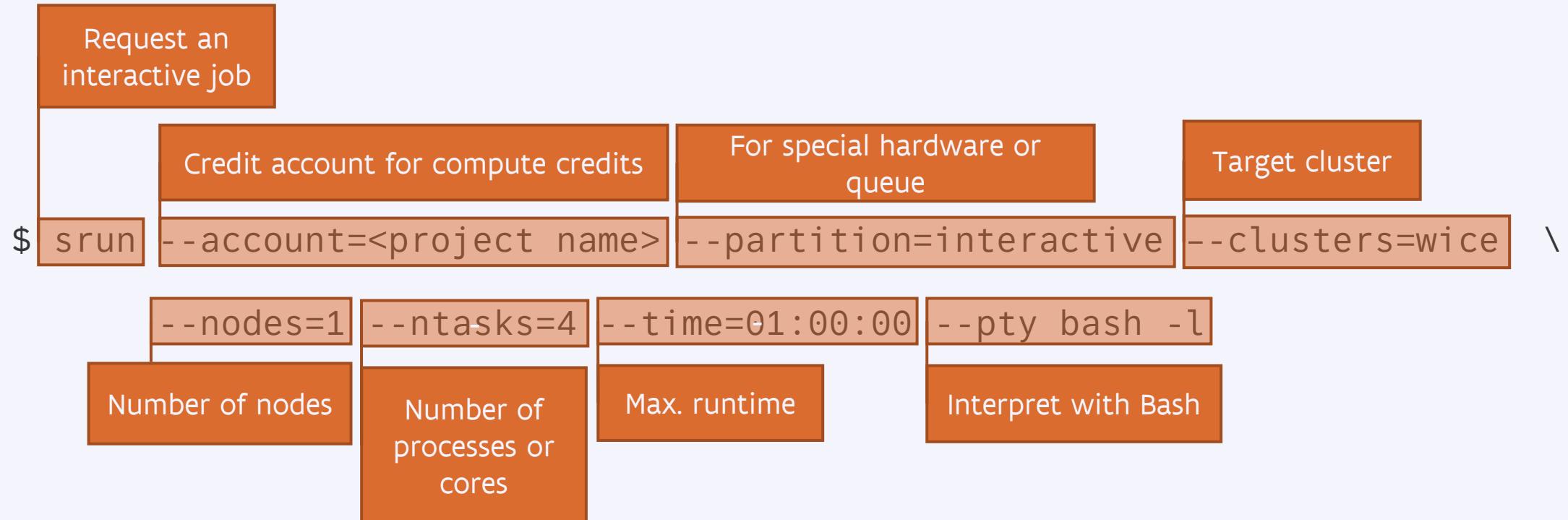
## Interactive job



## Batch job



# Example interactive job on wlCE



Remark

- Specifying <project name> for credits is mandatory, e.g. `-A lp_hpcinfo`
- Implicit defaults for an interactive job on wlCE are  
`-n 1 -t 01:00:00 -p batch --mem-per-cpu=3400M`

# Slurm command-line tools

Command	Purpose
\$ sbatch ...	Submit a batch job
\$ srun ...	Submit an interactive job
\$ scancel --cluster=wice <JobID>	Cancel a specific pending or running job
\$ scontrol show job --clusters=wice <JobID> \$ slurm_jobinfo <JobID>	Detailed job info (very useful to diagnose issues)
\$ squeue --clusters=wice --me --long	Status of all recent jobs
\$ squeue --clusters=wice --me --start	Give a rough estimate of when your jobs will start
\$ sinfo --clusters=wice	Info about the state of available partitions and nodes
\$ sacct --clusters=wice --batch --job <JobID>	Show the jobscript used for a given batch job
\$ slurmtop	Overview of the cluster
\$ sam-balance	Overview of all your available credit projects
\$ sam-list-allocations	Detailed overview of your credit allocation history

# Slurm job submit options

Slurm option	Remarks
--nodes=X --ntasks-per-node=Y --ntasks=X*Y --ntasks=X --cpus-per-task=Y	or or (e.g. Hybrid MPI + OpenMP)
--partition=<partition_name>	Default: "batch" partition
--mem-per-cpu=<size><M>	Min memory per core, e.g. 5000M (Genius)
--time=<dd-hh:mm:ss>	e.g. 1-12:30:00
--job-name=<job_name>	
--output=<file_template>	STDOUT; default="slurm-%j.out"
--error=<file_template>	Default: redirect to STDOUT
--mail-type=FAIL,BEGIN,END	
--mail-user=<email-address>	
--export=<ALL, key=value>	Additional variables to pass to the job
--account=<account_name>	Credit account (mandatory)

# Slurm job submit options

Slurm option	Remarks
--account=<account_name>	Mandatory (credits)
--partition=<partition_name>	Default: "batch" partition
--time=<dd-hh:mm:ss>	e.g. 1-12:30:00
--nodes=X --ntasks-per-node=Y --ntasks=X*Y --ntasks=X --cpus-per-task=Y	or or (e.g. Hybrid MPI + OpenMP)
--mem-per-cpu=<size><M>	Min memory per core, e.g. 5000M (Genius)
--job-name=<job_name>	
--output=<file_template>	STDOUT+STDERR; default="slurm-%j.out"
--mail-type=FAIL,BEGIN,END	
--mail-user=<email-address>	
--export=<ALL, key=value>	Additional variables to pass to the job

# Long and short options

Some (not all) of the sbatch, srun and salloc command line options also have a short version

Short version	Long version	Meaning
-A	--account	Slurm account name
-a	--array	Job array range
-M	--clusters	Cluster name
-c	--cpus-per-task	Number of cores per task (default: 1)
-d	--dependency	Job dependency (after, afterok, afterany, ...)
-i	--input	STDIN filename
-e	--error	STDERR filename
-o	--output	STDOUT filename
-t	--time	Maximum walltime
-N	--nodes	Minimum num nodes
-n	--ntasks	Maximum num tasks
-p	--partition	Partition name

# Interactive jobs

- ✓ Interactive job with default options (1 core for 1 hour on the *batch* partition)

```
$ srun --clusters=wice --account=lp_hpcinfo --pty bash -l
```

- ✓ Interactive job with X-forwarding

```
$ srun --clusters=wice --account=lp_hpcinfo --x11 --pty bash -l
```

- ✓ Interactive job with multiple cores, on the *interactive* partition

```
$ srun --clusters=wice --account=lp_hpcinfo --nodes=1 --ntasks=4 \
--partition=interactive --pty bash -l
```

- ✓ Interactive job with multiple cores and a GPU device

```
$ srun --clusters=wice --account=lp_hpcinfo --nodes=1 --ntasks=18 \
--partition=gpu --gpus-per-node=1 --pty bash -l
```

- ✓ To join one of your running jobs (connecting to the first compute node in the job's allocation):

```
$ srun -M wice --jobid=<JobID> --overlap -pty bash -l
```

# Example Slurm job script

```
#!/bin/bash -l
#SBATCH --clusters=wice
#SBATCH --account=lp_hpcinfo
#SBATCH --nodes=1
#SBATCH --ntasks=4
#SBATCH --mem-per-cpu=3400M
#SBATCH --job-name=hpc_workflow
#SBATCH --mail-type=BEGIN,END,FAIL
#SBATCH --mail-user=my.name@kuleuven.be

module load intel/2024a
module load Python/3.12.3-GCCcore-13.3.0
which python
cd $VSC_SCRATCH/projects/simulations
cp -r $VSC_DATA/input_data .

python modelling.py

cp -r output_data $VSC_DATA
rm -rf ./input_data ./output_data
```

Diagram illustrating the components of the Slurm job script:

- Shebang**: `#!/bin/bash -l`
- Resource list**: `#SBATCH` directives
- Loading modules**: `module load` commands
- Move data**: `cd`, `cp` commands
- Execute commands**: `python` command
- Move data**: `cp`, `rm` commands

# Batch job workflow

- ✓ Submit the job to the scheduler
- ✓ Receive a unique JobID
- ✓ Error and output files

Job script

```
#!/bin/bash -l  
...  
...
```

Submit command

```
$ sbatch simulation.slurm
```

JobID

Submitted batch job **60042478** on cluster  
wice

stderr, stdout

```
$ ls *.out  
slurm-60042478.out
```

# Output files

- ✓ STDERR and STDOUT are by default redirected to a single file:

slurm-<JobID>.out

- ✓ Contains job info, all errors and warnings, and printouts
- ✓ Worth going through (especially if it's a new kind of job)
- ✓ Address all warnings and errors (if you can)
- ✓ Typical error examples on the right

STDOUT + STDERR

```
$ ls slurm-* .out  
slurm-60042478.out
```

Out of memory

slurmstepd: error: Detected 1 oom-kill event(s).  
Some of your processes may have been killed by the  
cgroup **out-of-memory** handler.

Short walltime

slurmstepd: error: \*\*\* JOB 60042478 ON s28c11n2  
CANCELLED AT 2023-02-08T10:03:43 DUE TO TIME LIMIT  
\*\*\*

Low disk space

IOError: [Errno 122] Disk quota exceeded

# Output files

- ✓ Standard output and error channels can be redirected to other files:

```
#SBATCH --output ...
#SBATCH --error ...
```

```
$ ls slurm-* .out
slurm-60041238.out
```

stdout

Output file

```
SLURM_JOB_ID: 60033947
SLURM_JOB_USER: vscXXXXXX
SLURM_JOB_ACCOUNT: lp_wice_pilot
SLURM_JOB_NAME: testjob
SLURM_CLUSTER_NAME: wice
SLURM_JOB_PARTITION: batch
SLURM_NNODES: 1
SLURM_NODELIST: m28c30n4
SLURM_JOB_CPUS_PER_NODE: 72
Date: Tue Jan 10 17:02:04 CET 2023
Walltime: 00-01:00:00
=====
/apps/leuven/rocky8/icelake/2021a/
softwar
e/intel-compilers/2021.2.0/compile
r/2021.2.0/linux/bin/intel64/icc
cp: cannot stat '/apps/leuven/trai
ning/test': No such file or directo
ry
Hello World
```

Resource summary

stdout

# Job monitoring

Example

```
$ scontrol show job --clusters=wice 60049330
  JobId=60049330 JobName=f70.slurm
  UserId=vsc3... (253...) GroupId=vsc3... (253...)
  Account=lp_my_project QOS=lp_my_project
  JobState=PENDING Reason=QOSGrpBillingMinutes
  SubmitTime=2023-02-16T00:24:58 EligibleTime=2023-02-16T00:24:58
  Partition=batch NodeList= NumNodes=4-4 NumCPUs=288 NumTasks=288 CPUs/Task=1
  TRES(cpu=288,mem=979200M,node=4,billing=733
  MinCPUsNode=72 MinMemoryCPU=3400M
  WorkDir=/vsc-hard-mounts/leuven-data/3.../vsc3.../Odonate_SOM
  StdErr=/vsc-hard-mounts/leuven-data/3.../vsc3.../Odonate_SOM/slurm-60049330.out
  StdIn=/dev/null
  StdOut=/vsc-hard-mounts/leuven-data/3.../vsc3.../Odonate_SOM/slurm-60049330.out
```

Diagnosis

Why does my job remain in a pending state?

Check out the “Reason” on Slurm docs: [https://slurm.schedmd.com/resource\\_limits.html](https://slurm.schedmd.com/resource_limits.html)

# Other wICE partitions

GPU

```
#SBATCH --partition=gpu
#SBATCH --nodes=1
#SBATCH --ntasks=18
#SBATCH --gpus-per-node=1
```

Big memory

```
#SBATCH --partition=bigmem
#SBATCH --nodes=1
#SBATCH --tasks-per-node=72
#SBATCH --mem-per-cpu=28000M
```

# Interactive partitions

- ✓ Accessible via command line and Open OnDemand
- ✓ To quickly compile, test, debug your (parallel) application
- ✓ To pre- or postprocess your data (incl. visualizations)
- ✓ Short queue time
- ✓ 12 nodes on Genius: 36 cores/node, 192 GB mem
- ✓ 4 nodes on wICE: 64 cores/node, 512 GB mem, 1 physical GPU (split in 7 instances)
- ✓ Max resource per user: 8 cores, 1 GPU instance, 16 hour walltime
- ✓ Free of charge

Interactive job

```
$ srun --account=lp_myproject --clusters=wice \
    --partition=interactive --nodes=1 --time=16:00:00 \
    --gpus-per-node=1 --pty bash -l
```

# Debugging partitions

- ✓ Accessible via command line and Open OnDemand
- ✓ Quickly test if your (parallel) application works
- ✓ Short queue time
- ✓ Only one job at a time, max walltime: 1 hr

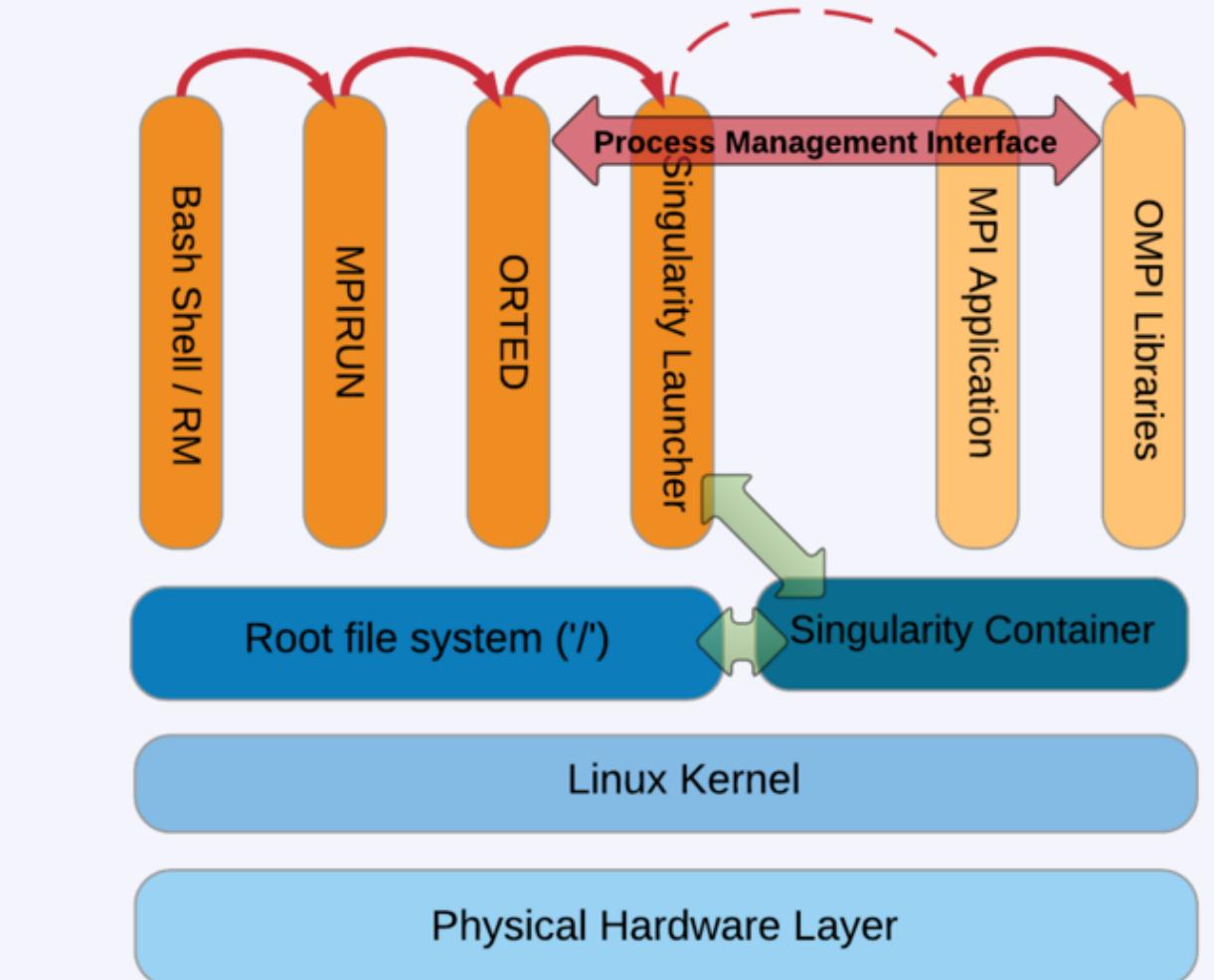
Cluster	Partition	Resources
Genius	batch_debug	2 Cascadelake nodes
	gpu_p100_debug	1 Skylake node with 4 P100 GPUs
wlCE	interactive	4 Icelake nodes, each with 7 MIG instances of 1 A100 GPU
	gpu_a100_debug	1 Icelake node with 1 A100 GPU

# Containers: Apptainer (Singularity)

- ✓ What?
  - Self-contained OS image in which you can install software and include data
- ✓ Why?
  - Package managers like apt can be convenient
  - Portability: images can be used on various hardware (but performance may be sub-par)
- ✓ How?
  - Create the image
  - Run it on wICE or Genius
  - Parallelization should work  
(multi-node MPI runs may require some care)

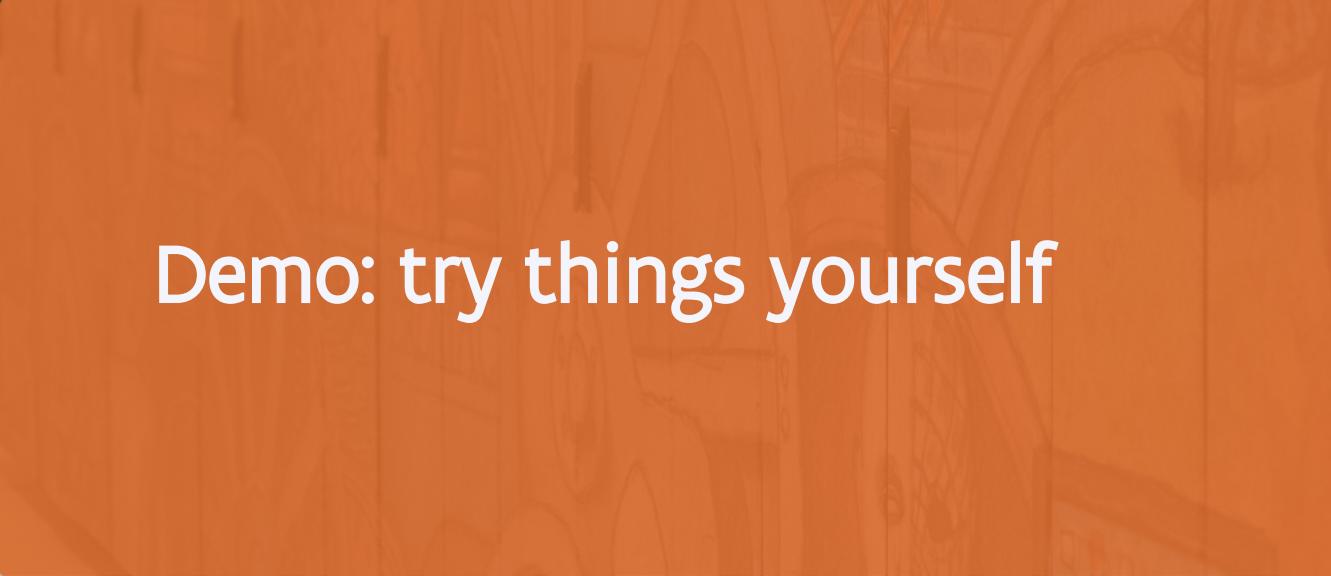
```
#!/bin/bash -l
#SBATCH --time=30:00
#SBATCH --nodes=1
#SBATCH --ntasks=72
#SBATCH --clusters=wice
#SBATCH --account=lp_hpcinfo
apptainer run Project.sif ./model.exe
```

script.slurm



# High-throughput computing with *atools*

- ✓ You can use atools to handle high-throughput computing workloads
- ✓ module load atools/<version>
- ✓ VSC training: Workflows for Data Science
- ✓ Training material: <https://github.com/gjbex/Workflows-for-HPC>
- ✓ Recording: <https://www.youtube.com/watch?v=GJwaRyi1LVA>
- ✓ User documentation: <https://atools.readthedocs.io/en/latest/>



Demo: try things yourself

# Demo: try things yourself

- ✓ Request to become a member of the */p\_hpcinfo* group via [account.vscentrum.be](http://account.vscentrum.be)
- ✓ Transfer a file with FileZilla
- ✓ Check your disk quota
- ✓ Check your available credits
- ✓ Check/load/list/unload/purge module

# Demo: try things yourself

- ✓ Copy `/apps/leuven/training/HPC_intro/` to your `$VSC_DATA`
- ✓ Submit `cpujob.slurm` to the cluster
- ✓ List all your jobs with `squeue -M wice`
- ✓ Check the information about the job with `slurm_jobinfo -M wice <job_ID>`
- ✓ Modify the `mpi.slurm` script to request 1 node, 72 cores for 30 minutes, with job start/end e-mail notifications
- ✓ Check the status of all the jobs

# Demo: job monitoring

## ✓ Submit an interactive job

Run your program on a compute node

Open a new terminal and ssh to the allocated compute node

Check the resources usage with htop

## ✓ Submit a GPU batch job

While the job is running, check its details with slurm\_jobinfo

and check resource on the compute node using ssh, htop and nvidia-smi

# Useful links

Helpdesk (email): [hpcinfo@kuleuven.be](mailto:hpcinfo@kuleuven.be)

Helpdesk (online form): [https://admin.kuleuven.be/icts/HPCinfo\\_form/HPC-info-formulier](https://admin.kuleuven.be/icts/HPCinfo_form/HPC-info-formulier)

VSC website: <https://www.vscentrum.be>

VSC documentation: <https://docs.vscentrum.be>

VSC agenda, training sessions, events (User Day): <https://www.vscentrum.be/vsctraining>

System status page: <https://status.vscentrum.be>

*Stay Connected  
to vsc*

