

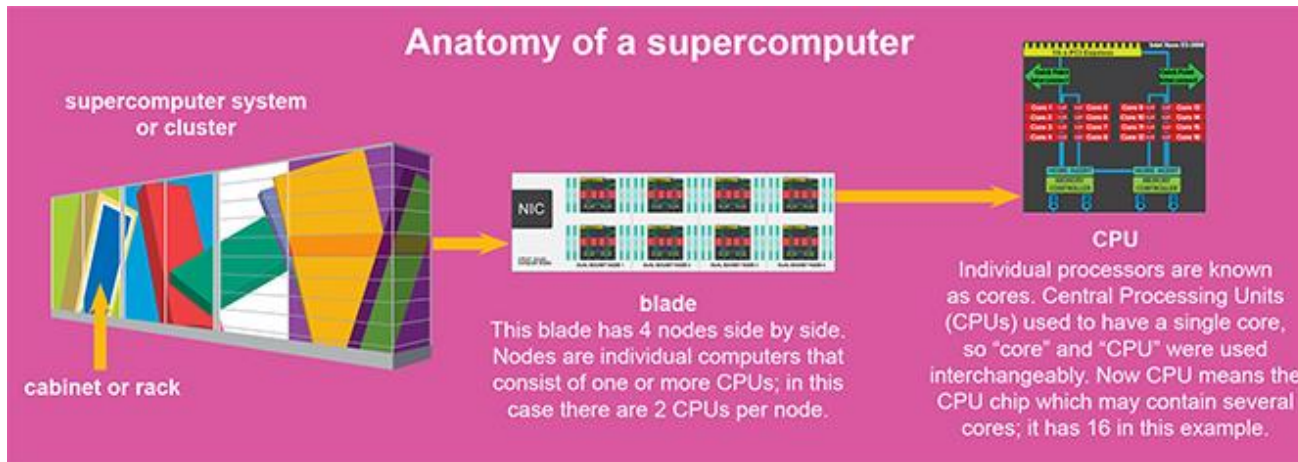


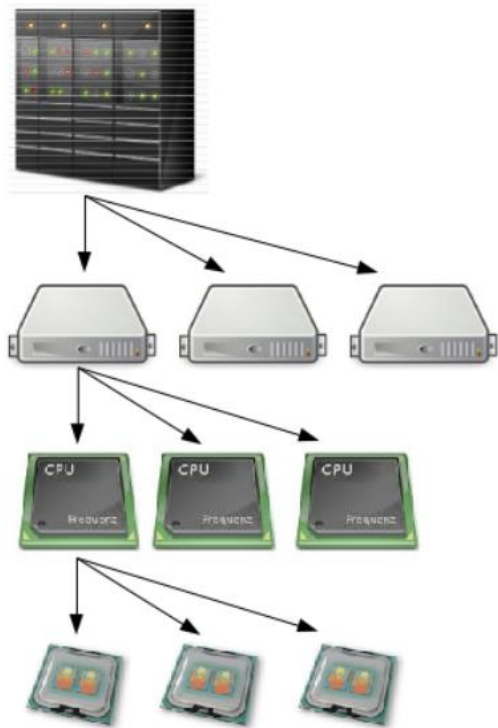
Introduction to HPC

PDC Summer School 2025

The architecture of a cluster

- Large system of interconnected compute devices (and storage)
- Hardware and software are tightly optimized for most efficient performance
- Maximization of throughput



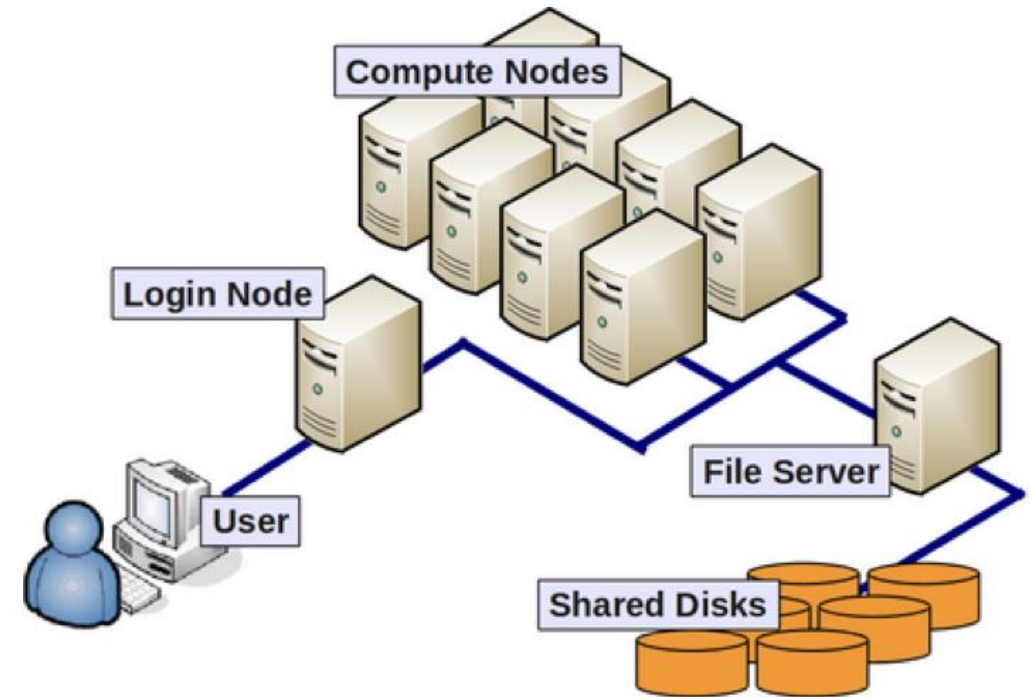


Cluster

Nodes

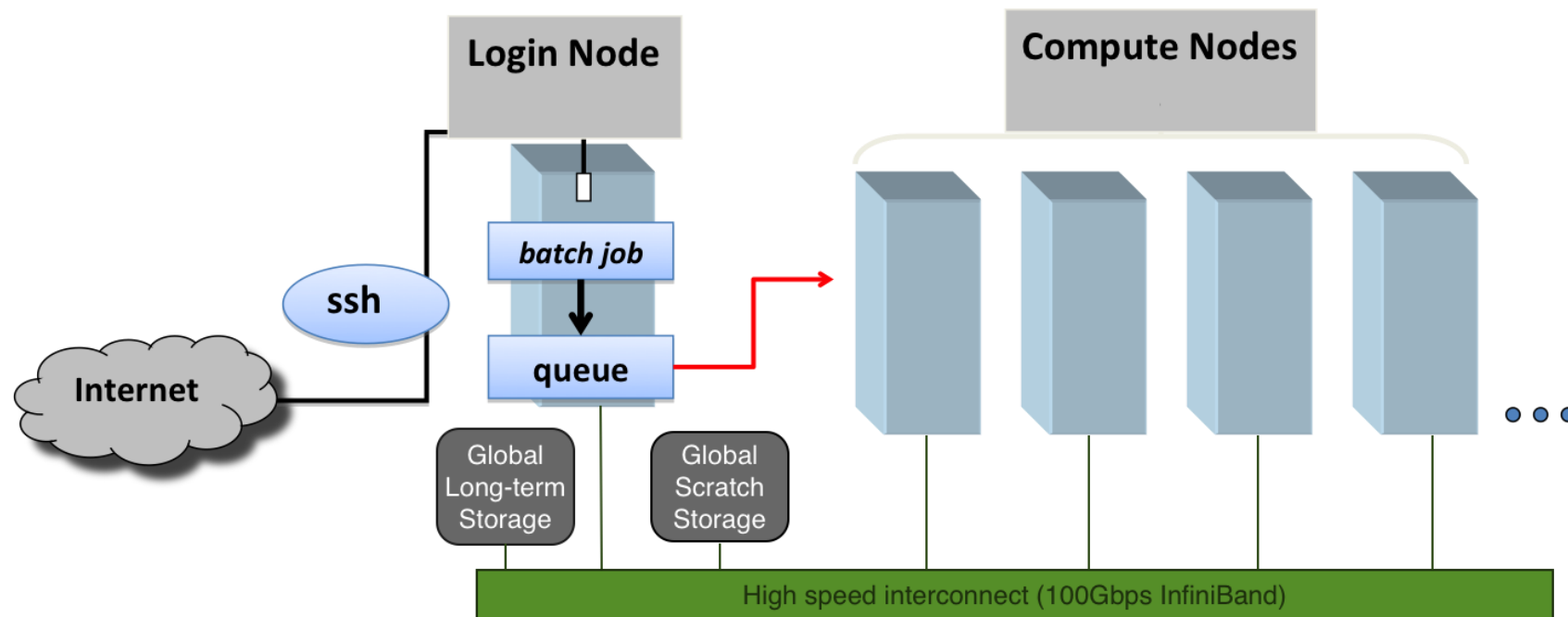
CPUs

Cores



<https://www.hpc.github.io/Introduction-HPC/101-introduction/index.html>

With great power comes great responsibility



<https://www.hpc.github.io/Introduction-HPC/101-introduction/index.html>

- Differences in:
 - Login node vs compute node
 - Loading and using software
 - 'Job' submissions and queueing due to shared usage

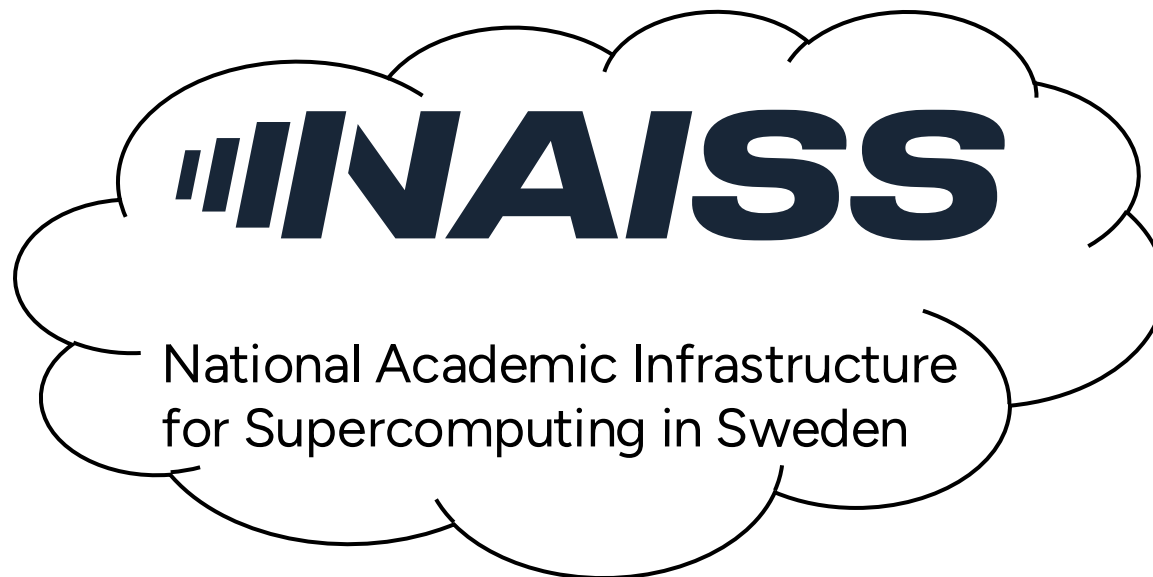


Introduction to Dardel

PDC Summer School 2025

Overview

- General information about PDC
- PDC resources & activities
- Access to PDC resources
- Login
- File systems, permissions and transfers
- Modules
- Running jobs
- Compiling programs
- How to contact PDC support



LUND UNIVERSITY

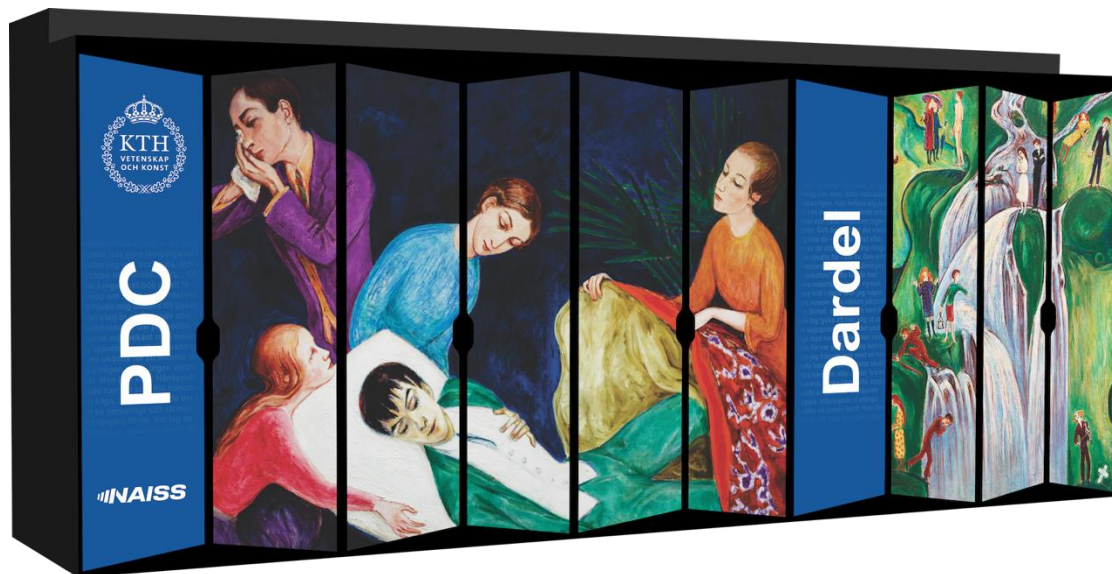
- Infrastructure organization for HPC in Sweden
- Primarily funded by the Swedish Research Council (VR)
- Integrates resources across HPC centres through partnerships with several universities



UNIVERSITY OF
GOTHENBURG

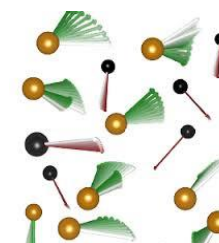


PDC Centre for High Performance Computing – Current Activities



- Operation of HPC + storage facilities
 - Dardel
 - PDC Cloud
 - Additional storage options, such as Tape
- Support for using NAISS resources
- Application experts in different scientific fields
- Workshops, training, courses
- Research collaboration with academia and industry
- International and national projects
 - Epicure
 - Ganana
 - BioExcel

- Cutting-edge software development
 - Neko (Computational Fluid Dynamics)
 - GROMACS (Molecular Dynamics)
 - Veloxchem (Quantum Chemistry)
 - VIAMD (Visualisation for Molecular Dynamics)
 - UPPASD (Material Science)



PDC Centre for High Performance Computing – History

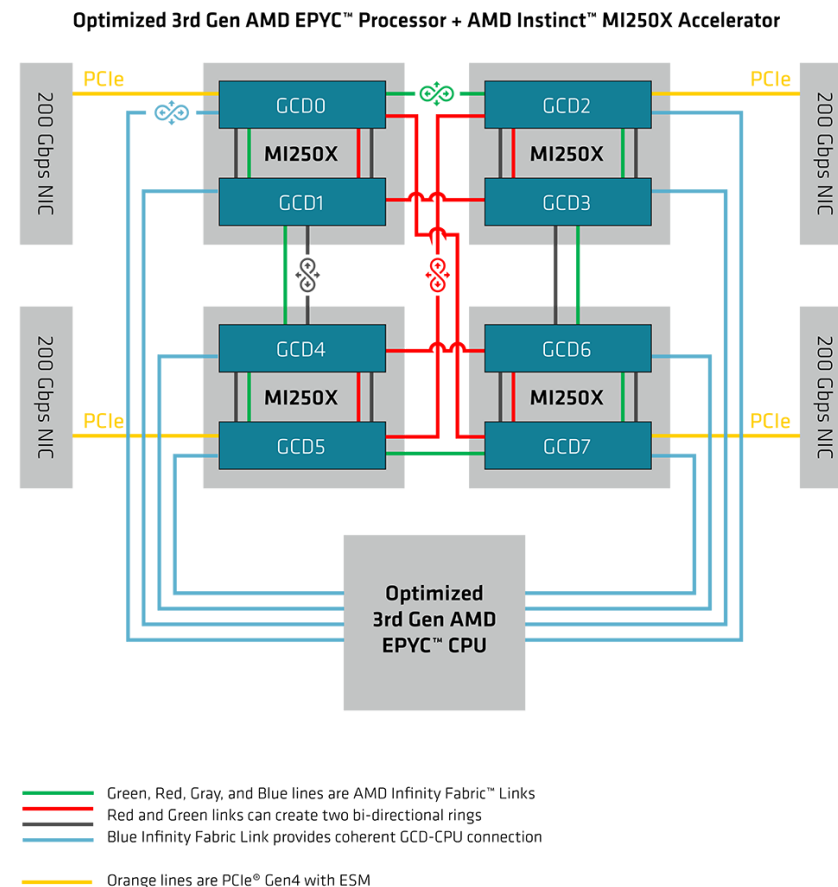
Year	rank	procs.	peak TFlops	vendor	name	remark
2023	153 (CPU) 77 (GPU)	163584	5055	Cray	Dardel	Cray EX 64-core 2.25GHz
2017	69	67456	2438.1	Cray	Beskow	XC40 16-core 2.3GHz
2014	32	53632	1973.7	Cray	Beskow	
2011	31	36384	305.63	Cray	Lindgren	XE6 12-core 2.1GHz
2010	76	11016	92.534	Cray	Lindgren	
2010	89	9800	86.024	Dell	Ekman	PowerEdge SC1435 Dual core Opteron 2.2GHz
2005	65	886	5.6704	Dell	Lenngren	PowerEdge 1850 3.2GHz

PDC Centre for High Performance Computing – History

2003	196	180	0.6480	HP	Lucidor	Cluster Platform 6000 rx2600 Itanium2 900MHz
1998	60	146	0.0934	IBM	Strindberg	SP P2SC 160MHz
1996	64	96	0.0172	IBM	Strindberg	
1994	341	256	0.0025	Thinking Machines	Bellman	CM-200/8k

Dardel

- CPU partition
 - 1278 nodes
 - Each node has two AMD EPYC (TM) Zen2 2.25GHz 64-core processors
- GPU partition
 - 56 nodes
 - Each node has
 - One AMD EPYC (TM) processor with 64 cores
 - Four AMD Instinct (TM) MI250X GPUs
- (Upcoming) Nvidia Grace-Hopper GPU
 - 8 nodes
 - Each node has
 - Four NVIDIA GRACE ARM CPUs
 - Four NVIDIA Hopper (H100) GPUs
- File system (Klemming)
 - Lustre file system
 - 23.6 PB





More about the CPU nodes

Number of nodes	RAM per node (GB)	Name
700	256	NAISS thin nodes
268	512	NAISS large nodes
8	1024	NAISS huge nodes
18	2048	NAISS giant nodes
36	256	KTH industry/business research nodes
248	512	KTH industry/business research nodes

Access to PDC resources

- PDC resources are free for Swedish academia
- Please acknowledge NAISS/PDC in your publications "The computations/data handling/[SIMILAR] were/was enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at [NAISS AFFILIATED SITE] partially funded by the Swedish Research Council through grant agreement no. 2022-06725"
- A project :
 - Can have allocations on multiple resources
 - Measure of how many jobs one can run (core-hours/month) and how much data one can store
 - Required for logging in
- Project allocations (compute & storage) are managed through SUPR/NAISS
<https://supr.naiss.se/round/>
- Flavours of allocations

	Small allocation	Medium allocation	Large allocation
Dardel (c-hrs/month)	20k	400k	>400k
Dardel – GPU (gpu-hrs/month)	200	4000	>4000
Klemming (GiB)	5k	100k	>100k
Eligibility (min.)	PhD student	Asst. Prof.	Senior Scientist
Evaluation	Weekly	Monthly	Half-yearly

Access to PDC resources

- Industrial users can also get access to PDC resources
 - Contact business-unit@pdc.kth.se or PDC support through regular channels (described towards the end)
- Some current business partners - <https://www.pdc.kth.se/industry/pdc-partners/business-partners-1.739819>

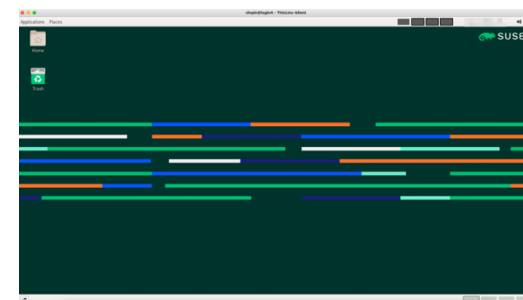
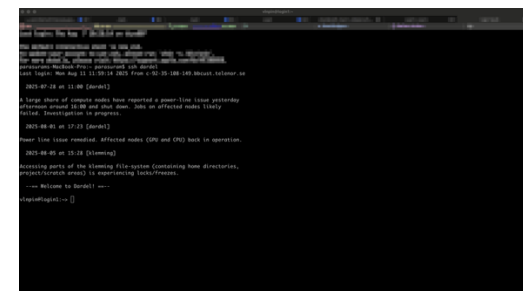
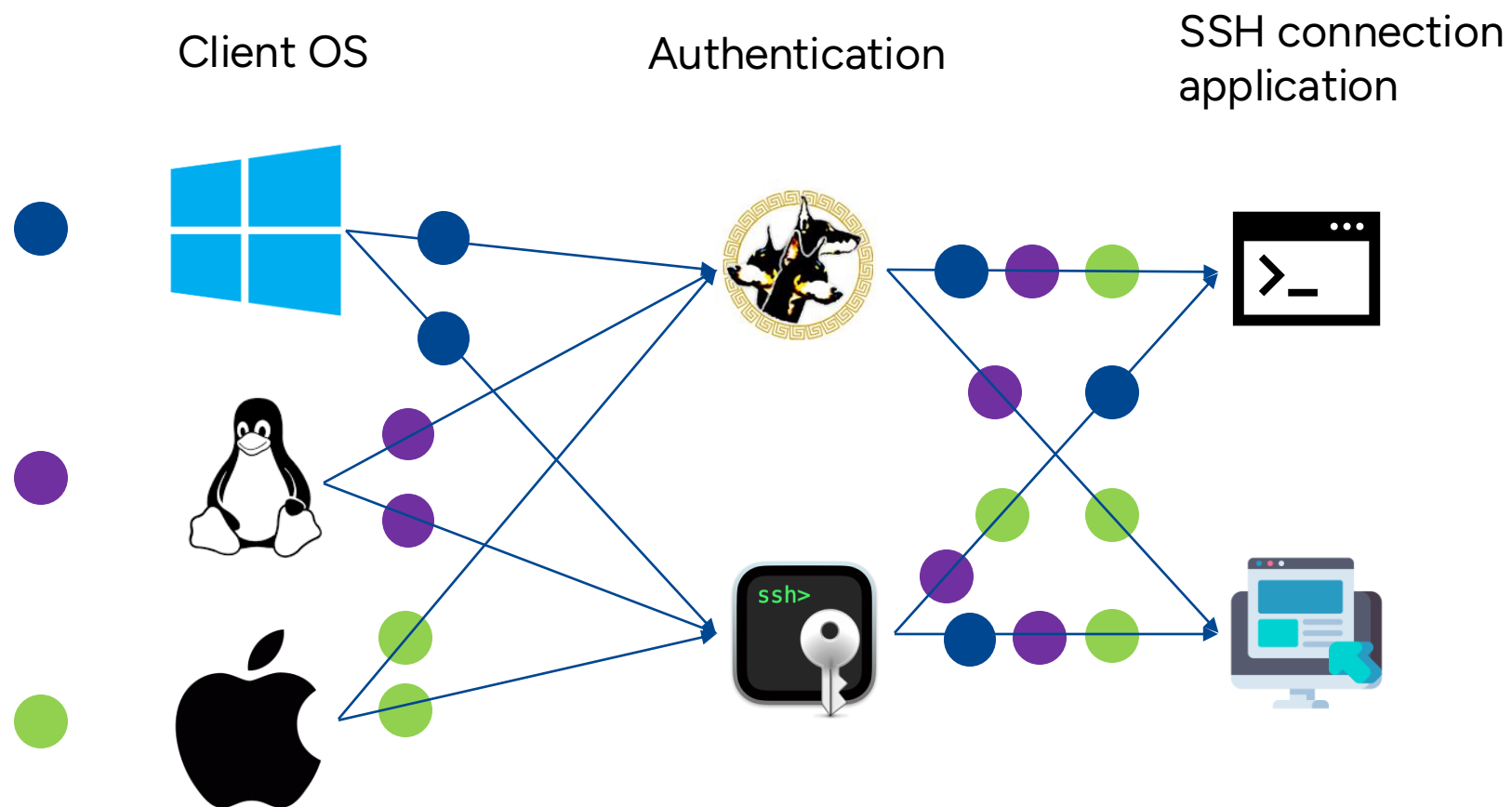


The figure shows a volume rendering of the instantaneous velocity magnitude on the leeward side of a Scania R20 Highline truck at crosswind conditions. (Source: Scania)



Login

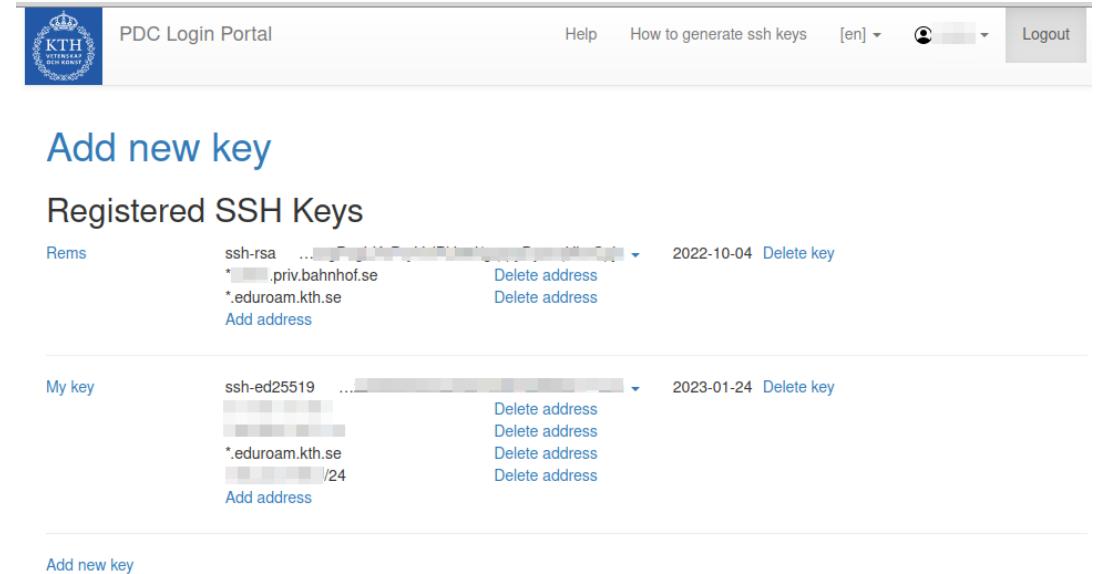
- Preliminary things to ensure
 - Membership in an active project allocation
 - PDC account
 - SSH client application – Terminal/Thinlinc
- Project allocation:
 - Mostly through SUPR
 - PI submits a proposal for resource allocation
 - PI can manage members in the project via SUPR
https://support.pdc.kth.se/doc/getting_access/get_access/#getting-compute-time
 - Course, workshop allocation are managed locally
https://support.pdc.kth.se/doc/getting_access/get_access/#request-class-access
- Request PDC account through
 - SUPR: For projects with an allocation granted through NAISS/SUPR
https://support.pdc.kth.se/doc/getting_access/get_access/#apply-via-a-supr-account
 - PDC Webform: Courses, training allocations
https://support.pdc.kth.se/doc/getting_access/get_access/#apply-via-pdc-webpage



Authentication methods



- Generate Kerberos tickets before logging in
- Subsequent logins don't require re-generation of the tickets as long as the ticket is valid
- Windows: Network Identity Manager + PuTTY or WSL
- Linux: Heimdal, openssh-client
- macOS: homebrew/openssh



PDC Login Portal

Help How to generate ssh keys [en] Logout

Add new key

Registered SSH Keys

Rems	ssh-rsa ...	2022-10-04	Delete key
	*.priv.bahnhof.se	Delete address	
	*.eduroam.kth.se	Delete address	
	Add address		

My key	ssh-ed25519 ...	2023-01-24	Delete key
	*.eduroam.kth.se	Delete address	
	/24	Delete address	
	Add address		

Add new key

- Generate SSH key-pair (private + public key)
- Register **public** key on the PDC login portal along with **all the IPs** you connect from
- Key registrations are valid for 365 days
- PDC account needs to be connected to SUPR account

Windows

- Kerberos

[PDC Support docs| Kerberos](#)

- Install Network Identity Manager for generating Kerberos credentials
- Install PuTTY for connecting via Terminal
- Ensure Kerberos tickets are valid/active before logging in

- SSH keys

[PDC Support docs| SSH keys](#)

- ⚠ Keypairs generated using PuTTY work for connecting via PuTTY, not other terminal applications
- ⚠ Only SSH key-pair based authentication works for connecting via Thinlinc on Windows
- ⚠ To use the SSH keys with Thinlinc, generate them using cmd/powershell, etc. and follow the instructions for linux

Linux/macOS

- Kerberos

[PDC Support docs| Kerberos](#)

```
$ kinit -f -l 7d <your-PDC-username>@NADA.KTH.SE  
$ ssh -o GSSAPIAuthentication=yes <your-PDC-username>@dardel.pdc.kth.se
```

⚠ GSSAPIAuthentication is set to yes by default in most cases, so it can be skipped

⚠ The Kerberos tickets must be generated on the local terminal.

⚠ Virtual/Conda environment

⚠ Conflicting installations, typically in macOS

- SSH keys

[PDC Support docs| SSH keys](#)

- Generate SSH keypair (not every time you connect)
- Register public key on the [PDC Login Portal](#) along with IP(s)
- Connect using the private key
- Pro tip: Use ssh config file

```
$ ssh-keygen -t ed25519 -f <file-path>
```

```
$ ssh -i <private-key path> <your-PDC-username>@dardel.pdc.kth.se
```



ssh config

- Manage login to multiple remote servers
- Manage authentication methods
- Create a file: ~/.ssh/config
- Example:

Host	dardel
HostName	dardel.pdc.kth.se
User	<your-PDC-username>
GSSAPIAuthentication	yes
IdentityFile	<private-key-path>

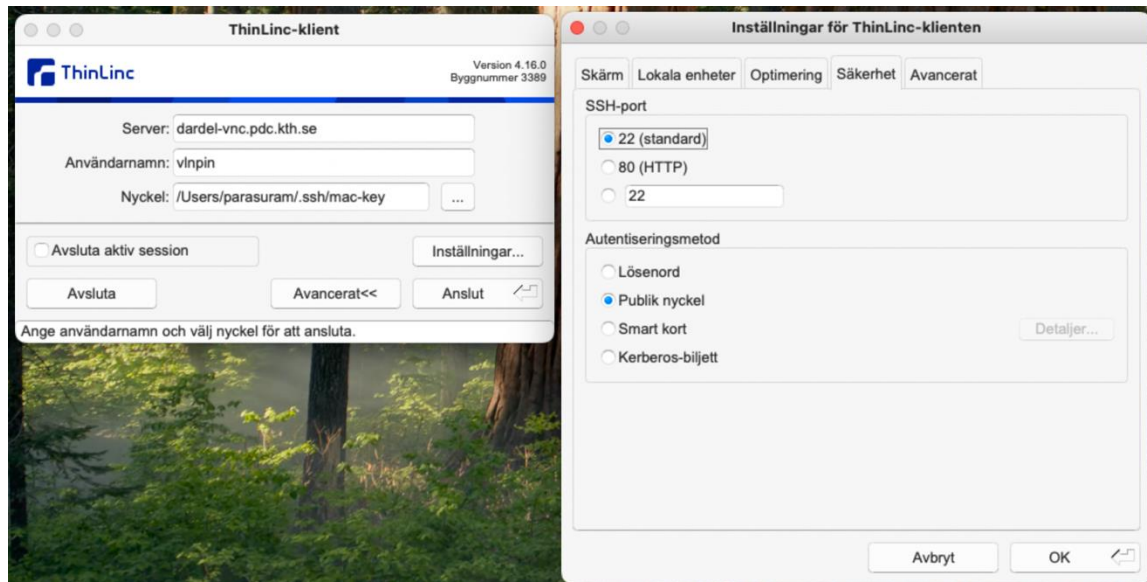
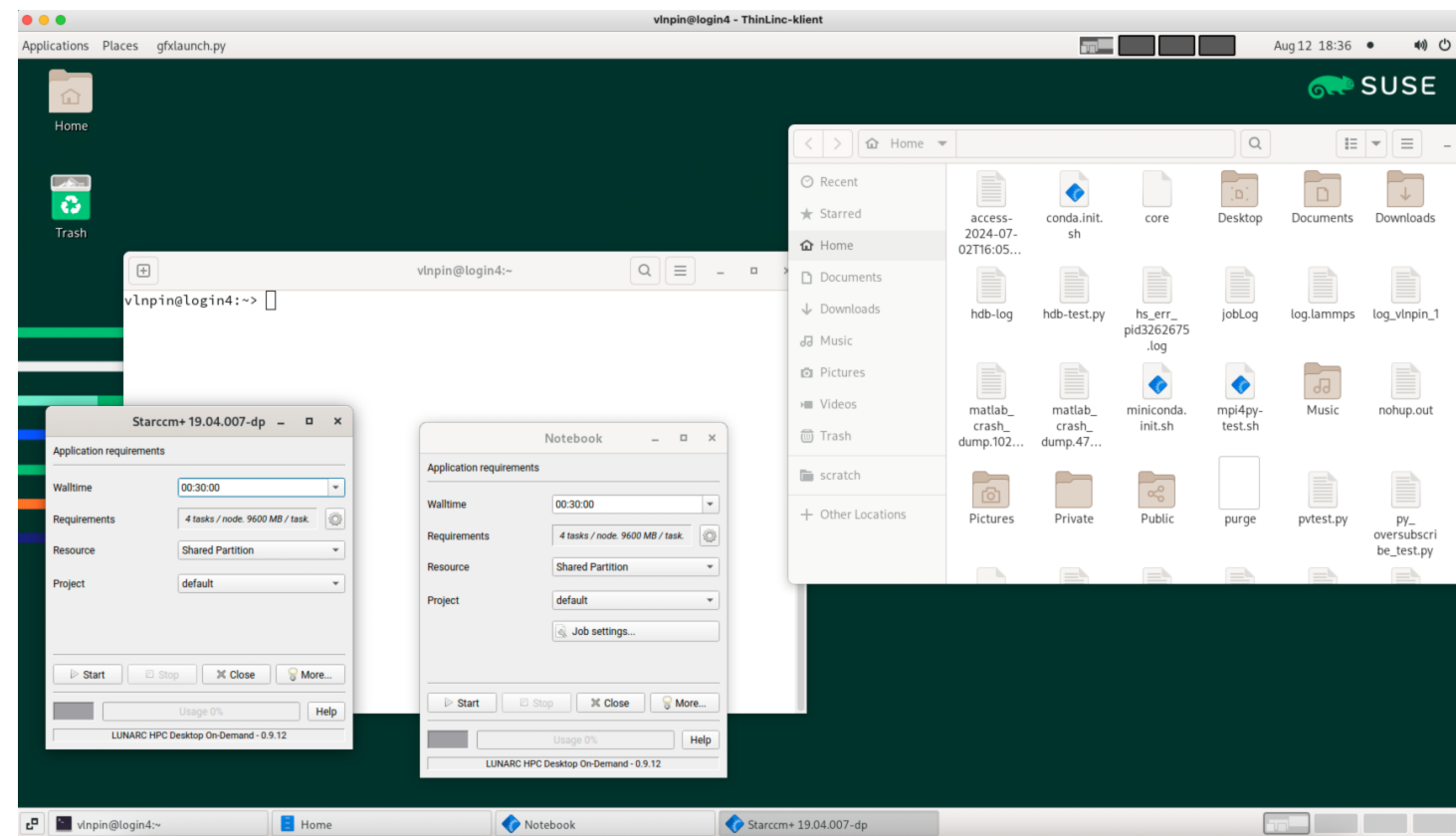
Host	server2
HostName	server2.blah.blah.se
User	<your-server2-username>

\$ ssh dardel



Thinlinc

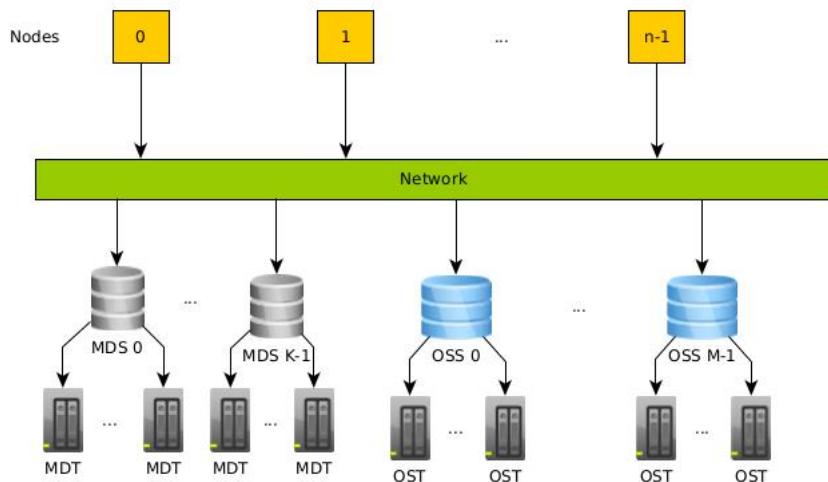
- Remote desktop environment
- GUI for many software
- Interactive job launcher



⚠ 'Logout' to release the license

[PDC Support docs| Thinlinc](#)

File system



CSC docs

(<https://docs.csc.fi/computing/lustre/>)

Area	Path	Alias	Size	File count	Backup
Home	/cfs/klemming/home/u/username	\$PDC_BACKUP	25 GB	100 K	Yes
Projects	/cfs/klemming/projects/snic/projectname	-	Varies	Varies	No
Scratch	/cfs/klemming/scratch/u/username	\$PDC_TMP	Unlimited	Unlimited	No

- Lustre File System – Klemming
- Open-source massively parallel distributed file system
- Optimized for handling data from many clients
- Use **projinfo** to check the project directory path as well as usage information
- ⚠ Note the quota on file count as well
- ⚠ (unmodified) scratch data is removed after 30 days



```
$ projinfo -h
```

```
Usage: projinfo [-u <username>] [-c <clustername>] [-a] [-o]
-u [user] : print information about specific user
-o : print information about all (old) projects, not just current
-m : print usage of all months of the project
-c [cluster] : only print allocations on specific cluster
-a : Only print membership in projects
-d : Usage by all project members
-p [DNR] : only print information about this project
-h : prints this help
```



File system

- Good practice
 - Minimize the number of I/O operations
 - Avoid creating too many files
 - Avoid creating directories with a large numbers of files
- Bad practice
 - Small reads
 - Opening many files
 - Seeking within a file to read a small piece of data

Access Control Lists (ACLs)

- Managing file permissions

```
$ getfacl -a /cfs/klemming/home/u/user/test

# file: /cfs/klemming/home/u/user/test
# owner: me
# group: users
user::rwx
group::r-x
other::---
```

← To view the access for a folder

To set access to another user →

To remove access →

To provide access to PDC support →

```
$ setfacl -m u:<uid>:r-x /cfs/klemming/home/u/user/test

$ setfacl -x u:<uid> /cfs/klemming/home/u/user/test

$ support-access <file/folder-name>
```

File transfer

- Same authentication methods as SSH
- rsync, scp, sftp, etc. can be used to transfer the files in both directions

⚠ Execute the file transfer commands on the local terminal for transfer in either direction

```
$ rsync <localfile> <username>@dardel.pdc.kth.se:<path>

$ rsync <username>@dardel.pdc.kth.se:<path> <localfile>

$ rsync <localfile> dardel:<path>

$ rsync -e "ssh -i <private-key> <username@>@dardel.pdc.kth.se:<path>" <localfile> <username>@dardel.pdc.kth.se:<path>
```

⚠ Use the file transfer node (*dardel-ftn01.pdc.kth.se*) for heavy file transfer processes

- If you have a ssh config setting, the command can be modified as
scp file.txt dardel:/cfs/klemming/scratch/[u]/[username]
- GUI based file transfer clients like WinSCP, Cyberduck, etc. can also be used. SSH key based authentication is supported by all of them

Modules

- Management of different software, versions, environments, etc
- Guess the number of modules available on Dardel
- Load installed software
- Install new software: Build/Compile, Easybuild, Spack

```
$ module show R/4.0.0
-----
/pdc/modules/system/base/R/4.0.0:

module-whatism GNU R
module-whatism
module          add gcc/7.2.0
module          add jdk/1.8.0_45
prepend-path    PATH /pdc/vol/R/4.0.0/bin
prepend-path    MANPATH /pdc/vol/R/4.0.0/share/man
prepend-path    LD_LIBRARY_PATH /pdc/vol/R/4.0.0/lib64/
-----
```

Command	Abbreviation	Description
module load [s]/[v]	ml [s]/[v]	Loads software/version
module avail [s]/[v]	ml av [s]/[v]	List available software
module show [s]/[v]	ml show [s]/[v]	Show info about software
module list	ml	List currently loaded software
module spider [s]	ml spider [s]	searches for software

[s]: Software. Optional for avail command
[v]: Version. Optional. Latest by default

Modules

- Some programs are optimized and therefore depend on the Cray Programming Environment (cpe)
- Unoptimized software can be loaded directly
- Optimized software is installed under the PDC module tree
- Each version of the PDC module corresponds to a cpe version
- For loading the modules of such optimized software, the **PDC** module needs to be loaded first
- [PDC Support docs](#) | [Software](#)

```
$ module avail PDC/
```

```
----- /pdc/software/modules -----  
PDC/24.11
```

If the avail list is too long consider trying:

"module --default avail" or "ml -d av" to just list the default modules.

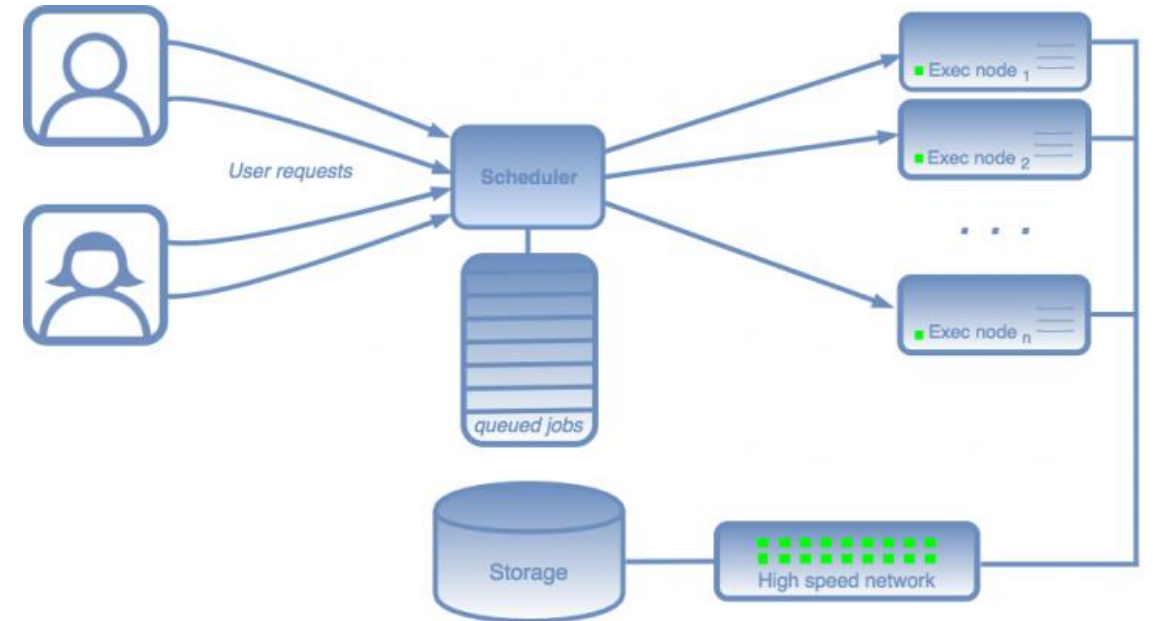
"module overview" or "ml ov" to display the number of modules for each name.

Use "module spider" to find all possible modules and extensions.

Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".

What next?

- ✓ Login
- ✓ Setup required input/case files
- ✓ Load the necessary software
- ✓ Build/compile your programs if necessary
- ✓ Run the simulation as a 'job' on a compute node





How to request support

- FAQs
[PDC Support docs| FAQs](#)
- The best way to get in touch is to contact us via our ticketing system
[PDC Support docs| Contact Support](#)
- Alternative option - email support@naiss.se
- NAISS Zoom-in
- Some good practices
 - Be descriptive !
 - Choose the appropriate tags when submitting the form through SUPR
 - Try to provide scripts, input files, etc. to reproduce the problem
 - Different topic – different thread