NSC introduction to Tetralith/Sigma

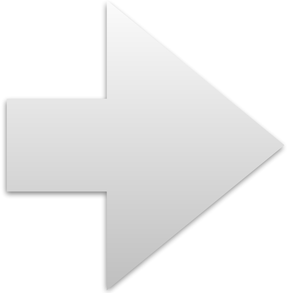
National Supercomputer Centre (NSC), Linköping University

NAISS training, online @NSC 15th Apr 2025, 10:00 - 12:00

LINKÖPINGS UNIVERSITET

NAISS

Information / Schedule



https://www.nsc.liu.se/support/Events/NSC_intro_Apr2025/

- this presentation as .pdf
- everything underlined is a link

10:00 Introduction to Tetralith/Sigma (Weine Olovsson)

~10:45 Using Python (Hamish Struthers)

~11:00 Using VSCode and Jupyter Notebook (Xuan Gu)

~11:15 Using GPUs (Torben Rasmussen)

~11:30 Open session, questions?

National Supercomputer Centre (NSC)

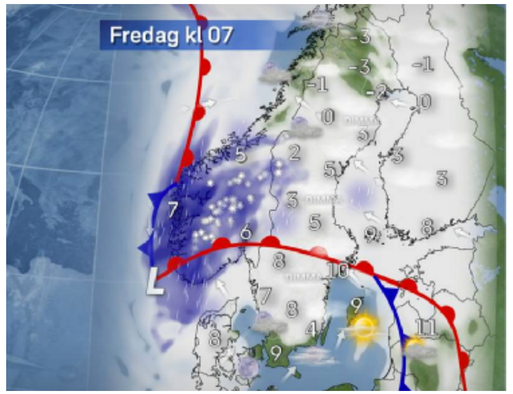
NSC is part of:

- **NAISS** National Academic Infrastructure for Supercomputing in Sweden (branch)
- LINKÖPING UNIVERSITY liu.se

1983 - SAAB buys Cray1

1989 - NSC first supercomputer centre in Sweden / SAAB partner

1996 - SMHI partner



2016 - MET Norway partner

https://www.nsc.liu.se

NAISS

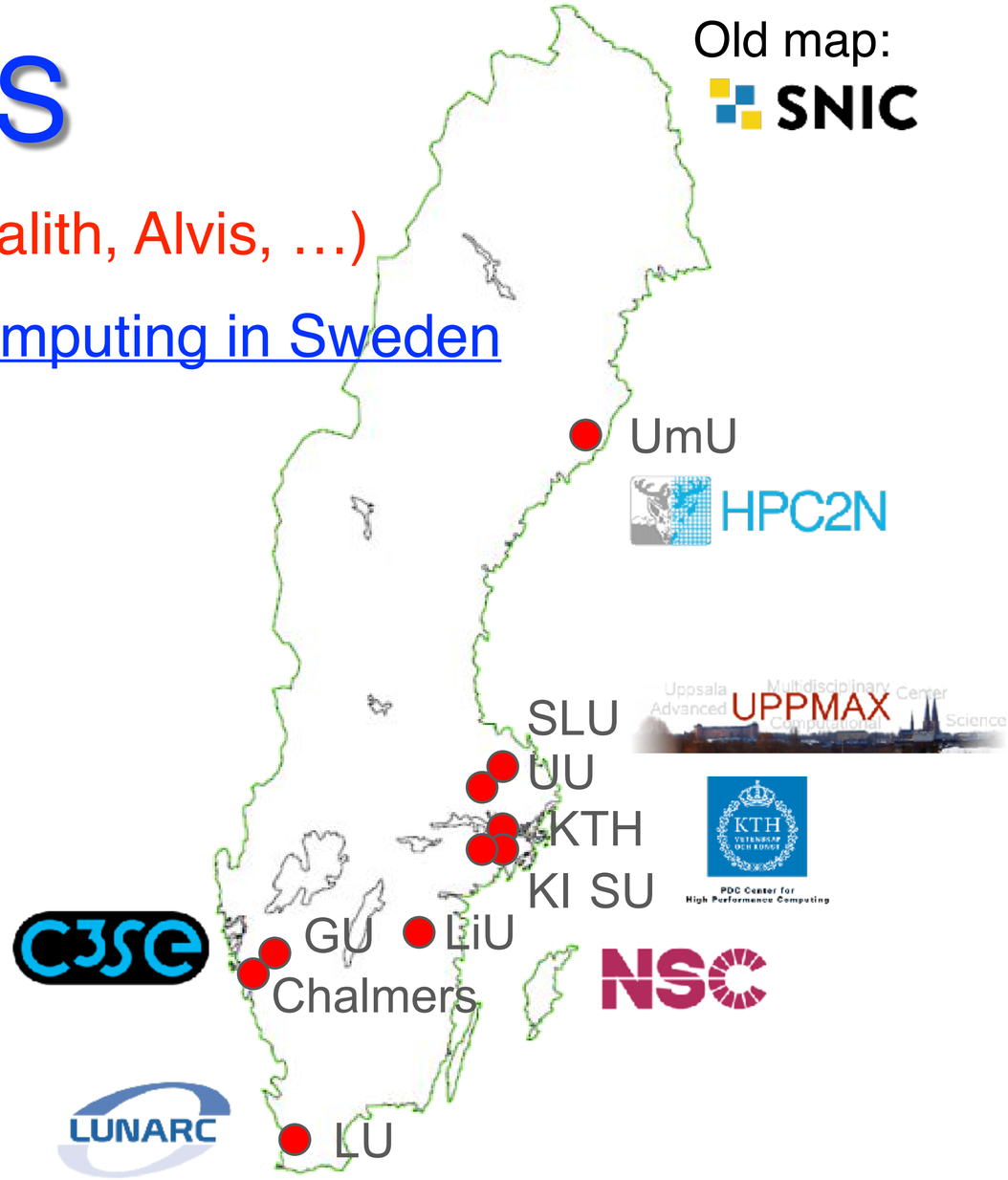
- Continues work of SNIC 2023 - (Dardel, Tetralith, Alvis, …)

National Academic Infrastructure for Supercomputing in Sweden

Branches at universities:

Chalmers	C3SE
Göteborg	
Karolinska	
KTH	PDC
Linköping	NSC
Lund	LUNARC
Stockholm	
Umeå	HPC2N
Uppsala	UPPMAX

+ more: SLU, LTU, LNU



https://www.naiss.se/ ← More info!

Funding: branches + Vetenskapsrådet (VR)

NAISS | The National Academic

https://www.naiss.se

NAISS

News Resources Allocations About / Om NAISS

National Academic Infrastructure for Supercomputing in Sweden

https://www.naiss.se/ More info!

NAISS and RISE to set up Swedish AI Factory

EuroHPC JU (Joint Undertaking) has approved a joint application from NAISS and RISE to set up an AI Factory in Sweden. The Factory, which will be called Mimer, combines a new AI-optimised supercomputer with comprehensive support for training, research

NAISS Resources

OVERVIEW
ALVIS
BIANCA
DARDEL
LUMI SWEDEN
SWEDISH SCIENCE CLOUD (SSC)
SWESTORE/DCACHE
TETRALITH

COMPUTE RESOURCE

Alvis

Host Data Centre

C3SE, Chalmers University of Technology

Purpose

Allocations

NAISS LARGE COMPUTE SPRING 2025 CALL
NAISS LARGE STORAGE SPRING 2025 CALL
LARGE CALLS 2024 >

NAISS coordinates and provides large-scale computing and storage capacity to researchers at Swedish universities and other research institutes authorised by the Swedish Research Council >. Resources are made available through rounds with open application procedures to ensure that the best Swedish research is supported.

Proposals are prepared and submitted via the Swedish User and Project Repository (SUPR) > platform.

Open rounds

NAISS resources are made available through a number of rounds with open application procedures. Computational and storage resources can be applied for as Small, Medium and Large projects.

STAFF
STEERING COMMITTEE
STEERING DOCUMENTS
ANNUAL REPORTS
NEWSLETTER
NEWS
EVENTS
TRAINING
CONTACT

About us

(För information om NAISS på svenska se denna sida >.)

The National Academic Infrastructure for Supercomputing in Sweden (NAISS) maintains advanced computing, artificial intelligence, and data services that support research ranging from fundamental research in mathematics or physics to climate modelling, the green transition, life science and economics. NAISS is hosted by Linköping University >, with funding provided by the Swedish Research Council (Vetenskapsrådet) > for the years 2023-2026 under contract 2022-06725.

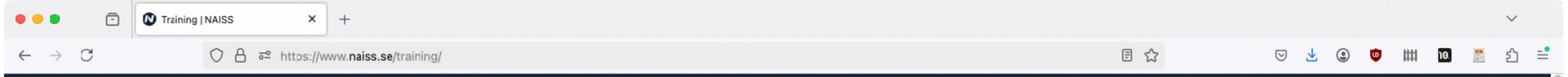
NAISS is the largest and most widely used research infrastructure in Sweden, involving more than 7,600 users at some 30 universities and research institutes (statistics from 2023). We also maintain special dedicated resources for infrastructures, agencies, and external users.

We aim to evolve with our users and their needs. NAISS should be your partner from the first time you open an account on a cluster and need Linux training to the point where you manage a team running a state-of-the-art simulation or AI training on 10,000 GPUs.

A diverse portfolio of computing infrastructure

The main resources provided by NAISS and collaborating organisations include

- High-performance computing clusters/supercomputers consisting of nodes with high-end CPUs or GPUs and large amounts of memory, where thousands of such nodes are connected by very fast networks.



NAISS News Resources Allocations About / Om NAISS

Training

NAISS offers a wide range of courses, workshops and seminars that are free to all users. Training is provided by staff from the NAISS branches distributed across Sweden.

This list is updated weekly.

For more information and registration, please click the titles below to go to the hosting branch event pages.

Date	Event	Where
15 April	NSC introduction to Tetralith/Sigma	Online
24-25 April, 28-29 April	Using Python in an HPC environment	Online
5-9 May	Programming Formalisms	Online
8 May	NAISS Zoom-in - a virtual open-house	Online
16 May	Transferring Files to/from HPC Clusters	Online
21 May	Introduction seminar for Alvis users	Online
	- Improve Your Handling of	Online
27 May	Cluster architecture and job submission	Online
2-3 June	Intermediate Bash and Linux	Online
10-11 June	Introduction to Pandas for data science	Online
12 June	NAISS Zoom-in - a virtual open-house	Online

- NAISS training info, also see training newsletter!

https://www.naiss.se/training

STAFF
STEERING COMMITTEE
STEERING DOCUMENTS
ANNUAL REPORTS
NEWSLETTER
NEWS
EVENTS
TRAINING
CONTACT

Access and support

All NAISS resource allocations are based on proposals. Getting started with a small allocation only requires a brief description of your needs in the SUPR user portal >, and we will typically get back to you within days. Medium and large-scale proposals undergo more extensive review, as described under Allocations >, where you can also find information about the National Allocation Committee (NAC). Computing and storage resources are typically scarce, and the prioritisation is based on scientific excellence and the expected impact.

Thanks to additional funding by our partner universities, NAISS is providing nationwide support through local staff at branches across Sweden, ensuring access to a range of different competences. Open house events will be organised at all these sites. In cases where the competence you need is not available locally, NAISS support staff will connect you with suitable experts at other sites. In some cases, local NAISS staff is physically co-located at a local computing center.

Governance

Major decisions in NAISS are taken by the Steering Committee, which is appointed by Linköping University Vice-Chancellor Jan-Ingvar Jönsson. Professor Jan-Eric Sundgren. The main executive officer of NAISS is the Director, Erik Lindahl, joined by Anna Jänis as head of administration and a management team including associate directors as described on the Staff > page.

- For access and support:

https://supr.naiss.se/

New NAISS cluster: Arrhenius

- Carl Axel Arrhenius (1757-1824), ytterbite discovery
- **EuroHPC “mid-range”**, ca. 6-700 MSEK
- Installation & start HT 2025 (CPU first, GPU later)
- “Tetralith replacement”
- Users: academic (EuroHPC/NAISS), industry

NSC: Quick Overview

Part of LiU and NAISS (Director: Erik Lindahl)

~ **40** people (not all full-time)

Mostly **system experts** and **application experts**

- Provide computational resources
- Software installation (global / local)
- Troubleshooting / advice
- Training (NAISS, local and other)

NSC Academic Clusters

32 cores/node

Tetralith (2018 -) 1908 x 2 x 16 cores, Intel Xeon Gold 6130

NAISS (2020 -) 170 x T4 GPU-nodes

Top500 no. 359 (74)

Sigma (2018 -) 110 x 2 x 16 cores, Intel Xeon Gold 6130 *"same" as Tetralith*

(2020 -) 2 x V100 GPU-nodes

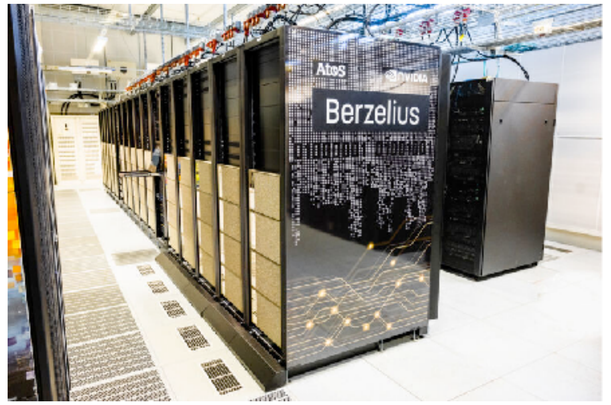
LINKÖPING UNIVERSITY

BerzeLiUs (2021 -) Nvidia DGX SuperPOD, (60 + 34) x 8 A100 **GPUs**

LINKÖPING UNIVERSITY

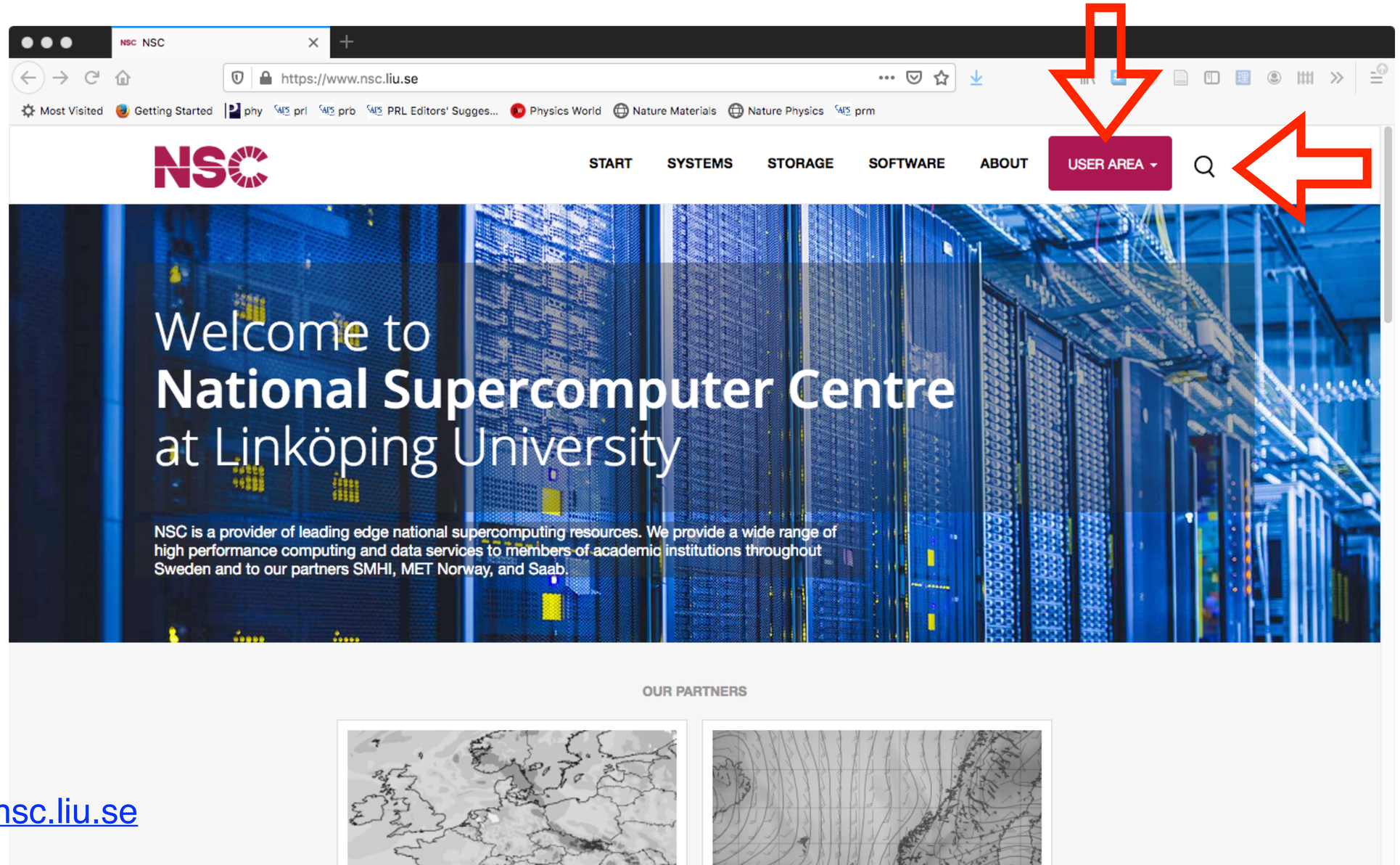
Knut and Alice Wallenberg Foundation

AI/ML, for all researchers in Sweden



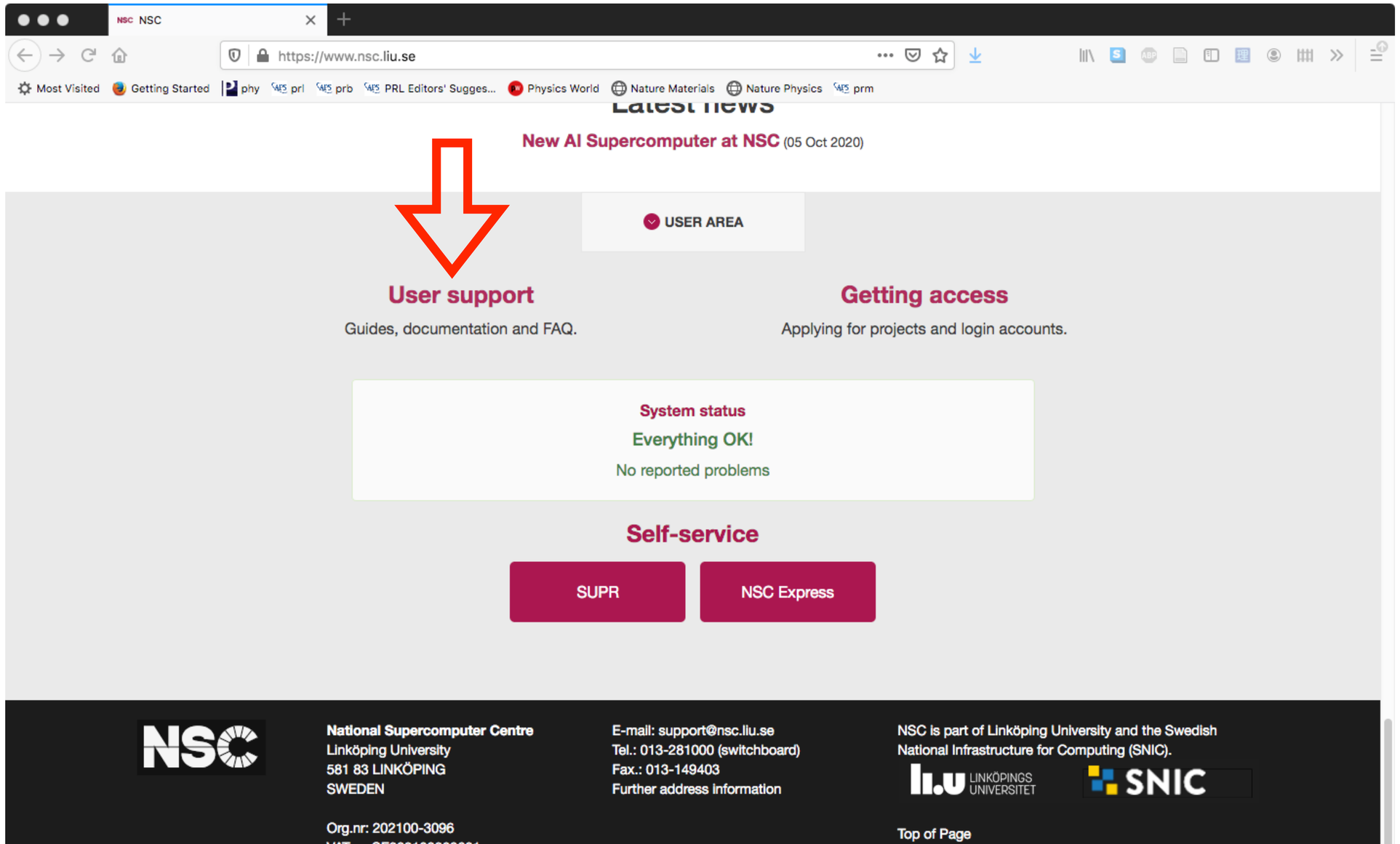
Top500 no. 204 & 237 (83)

Where to find Information?

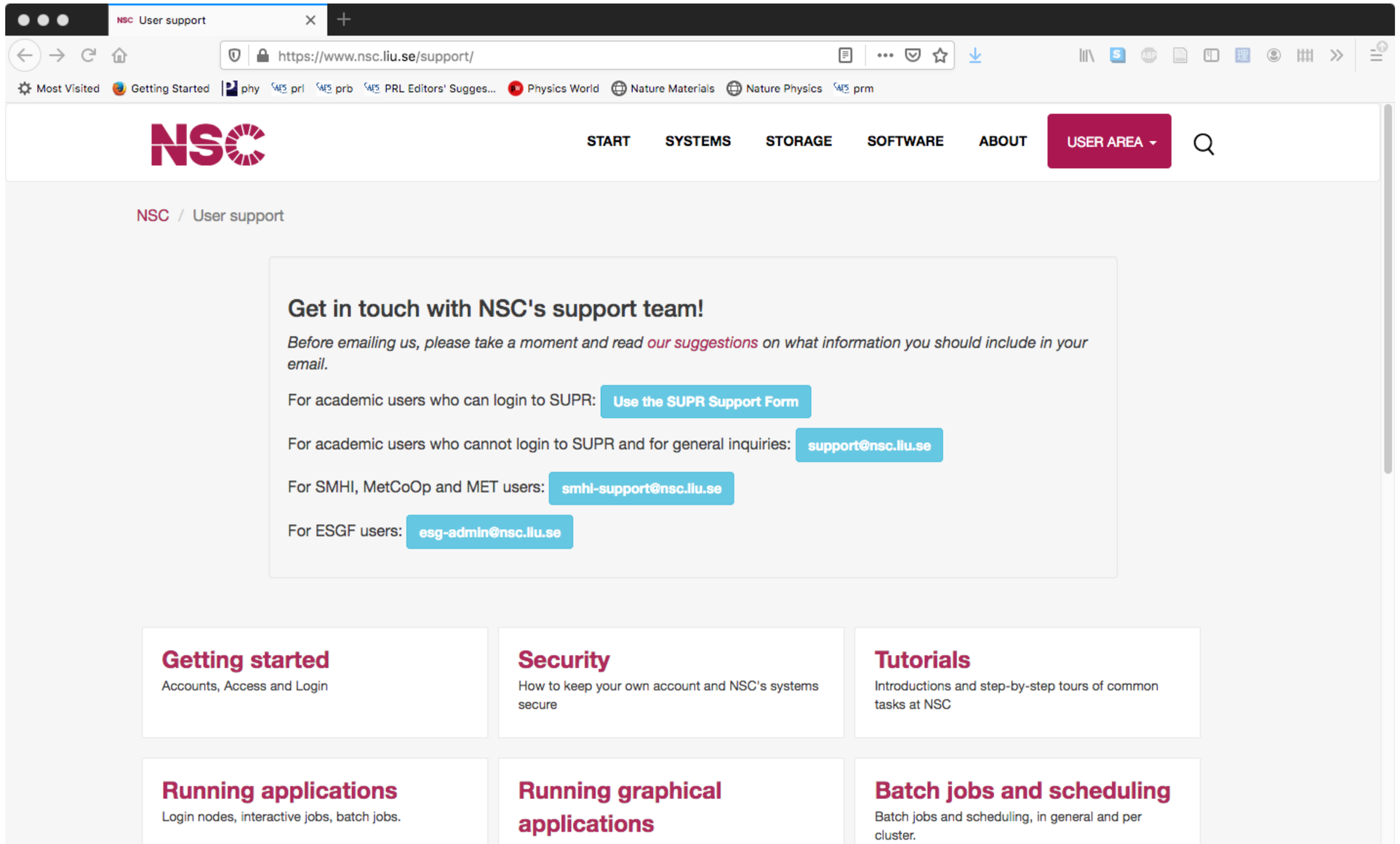


https://www.nsc.liu.se

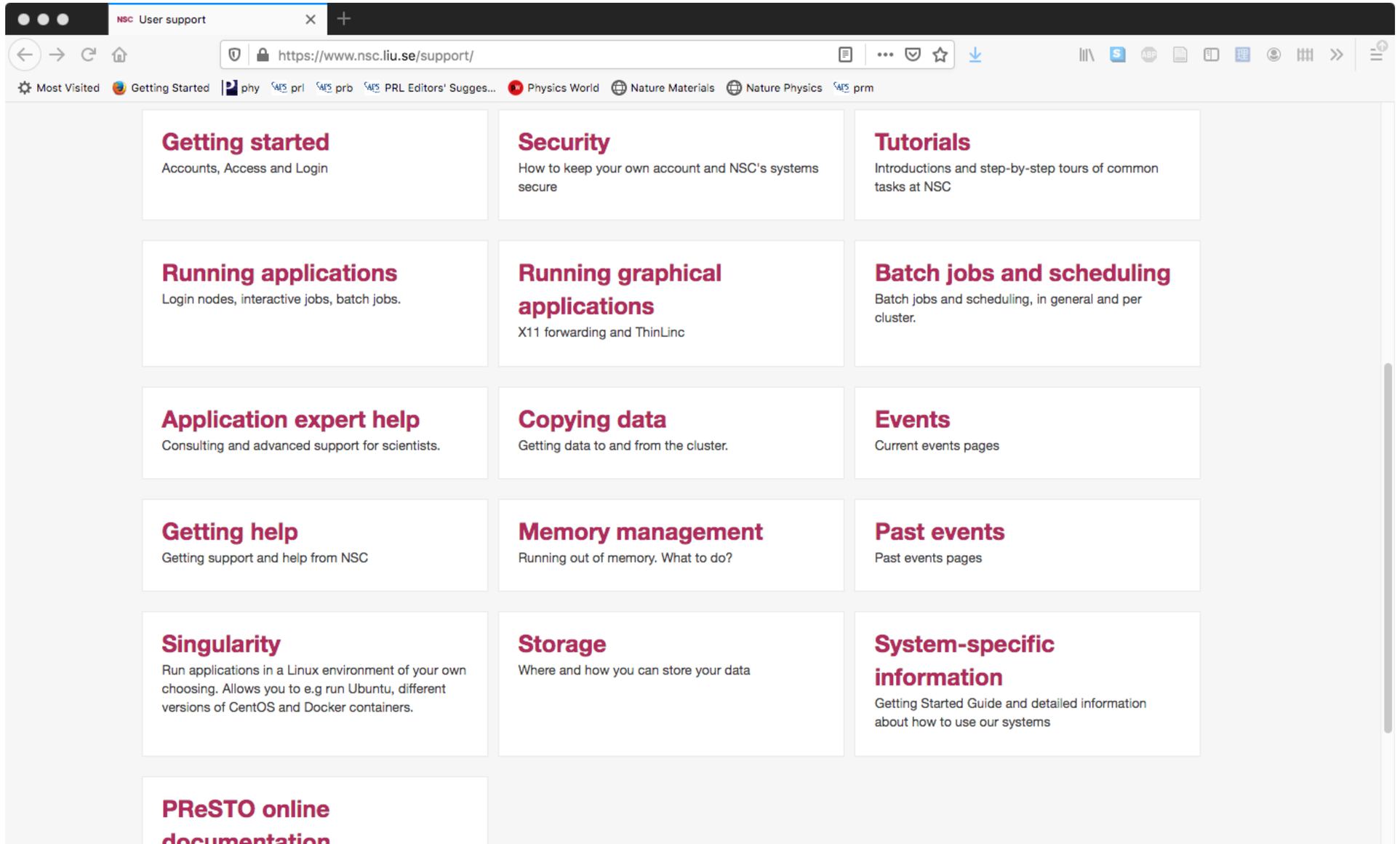
Where to find Information?



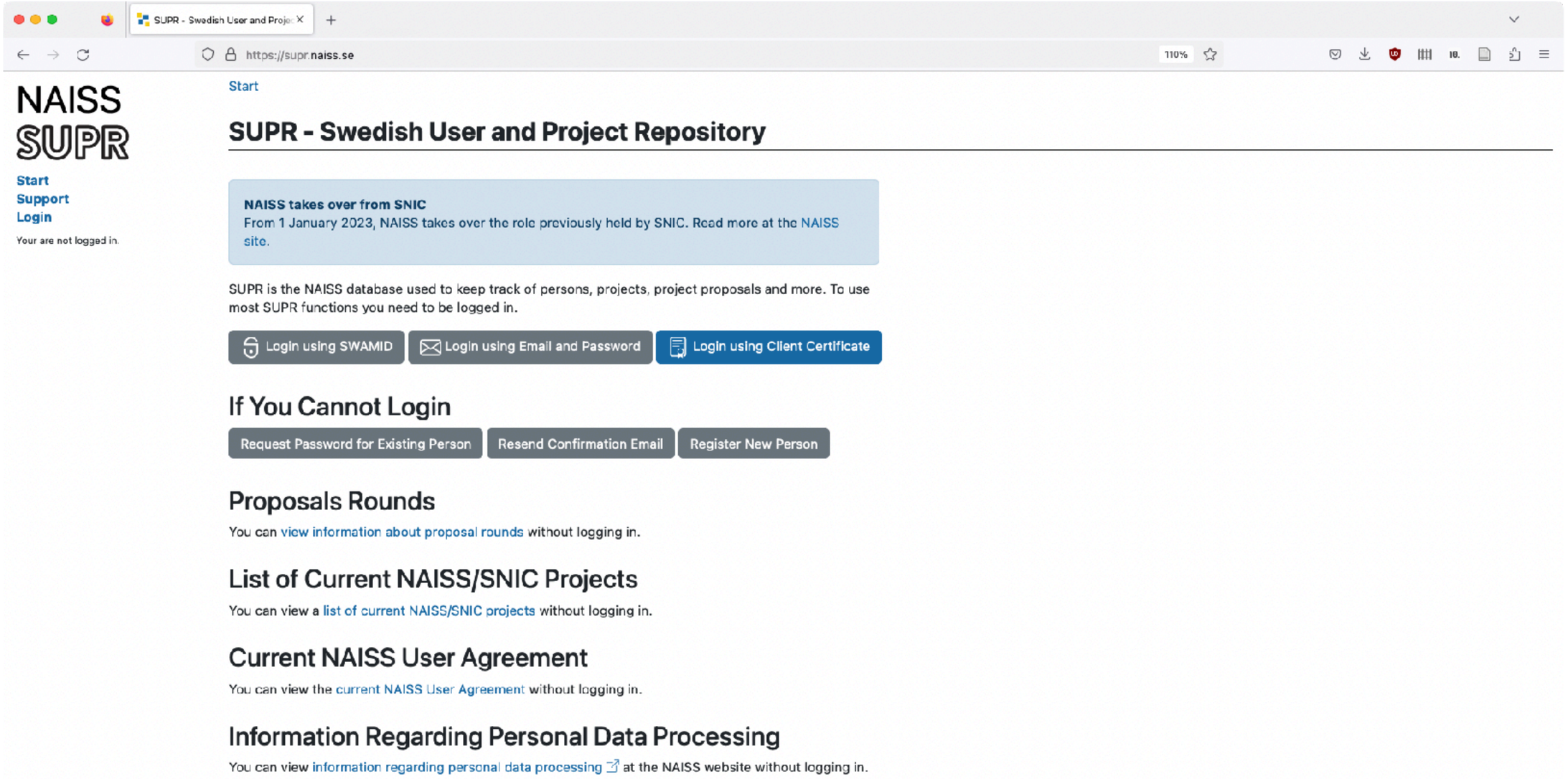
Where to find Information?



Where to find Information?

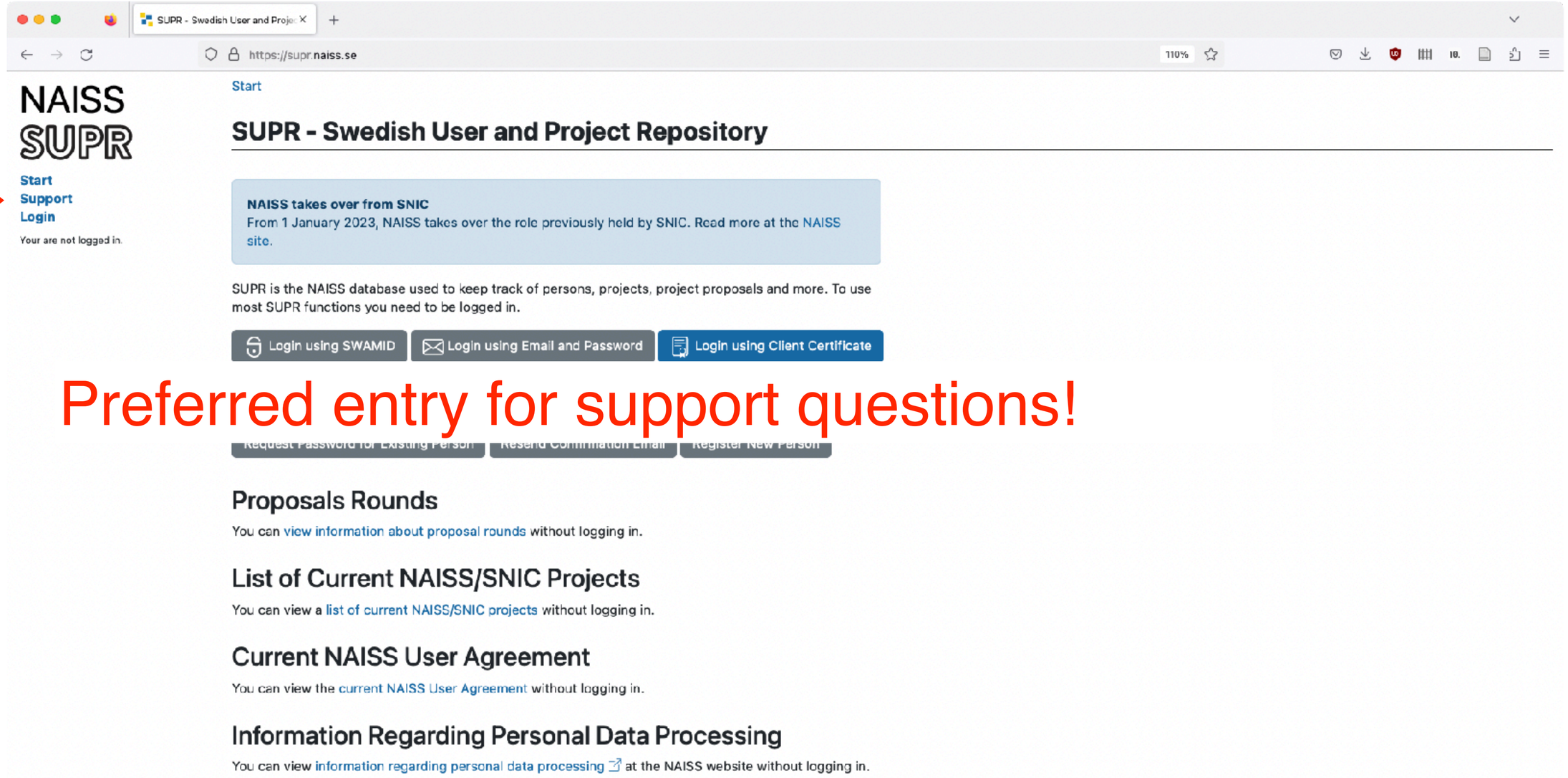


Getting Acess to HPC - SUPR



https://supr.naiss.se/

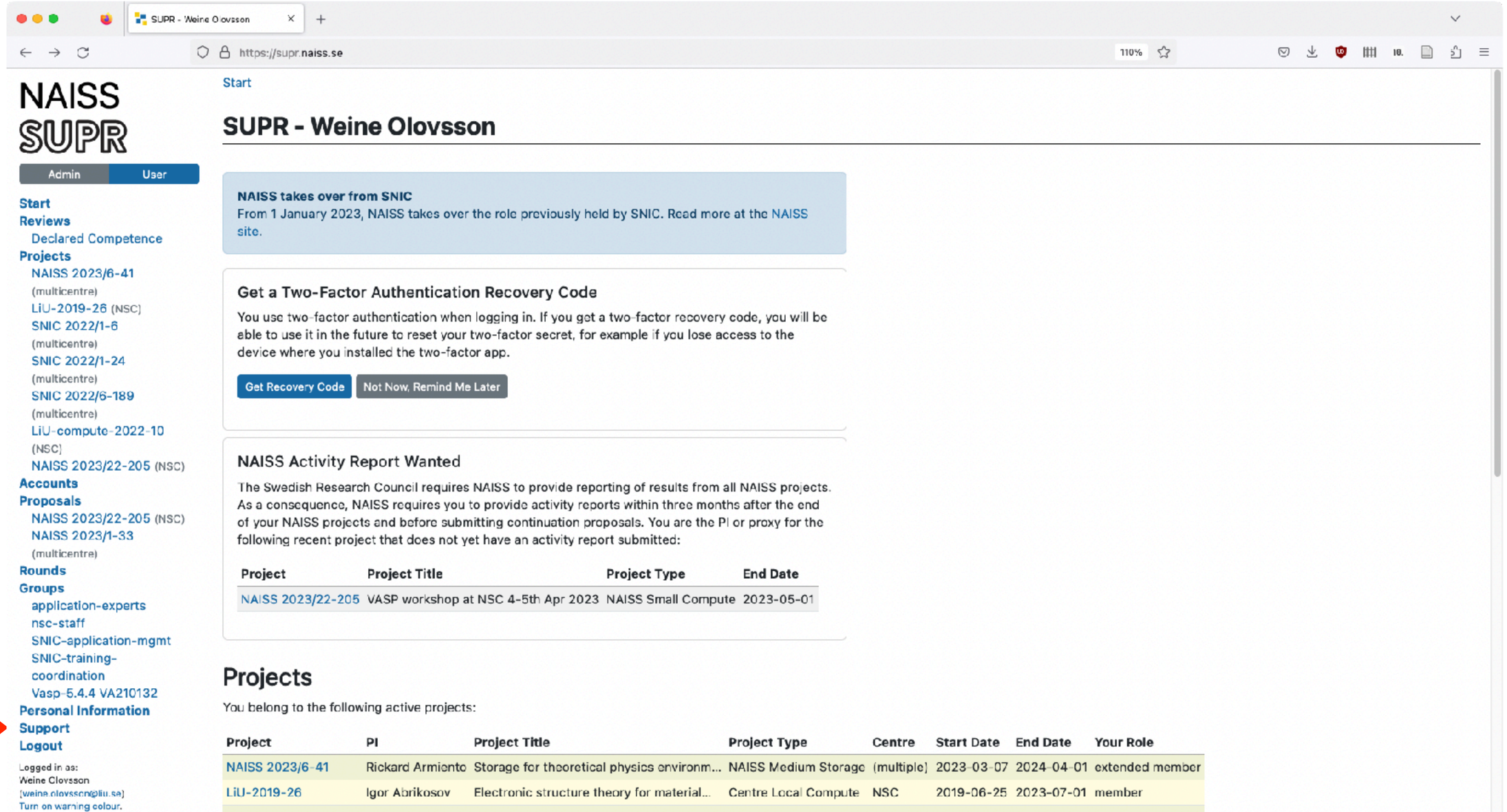
Getting Acess to HPC - SUPR



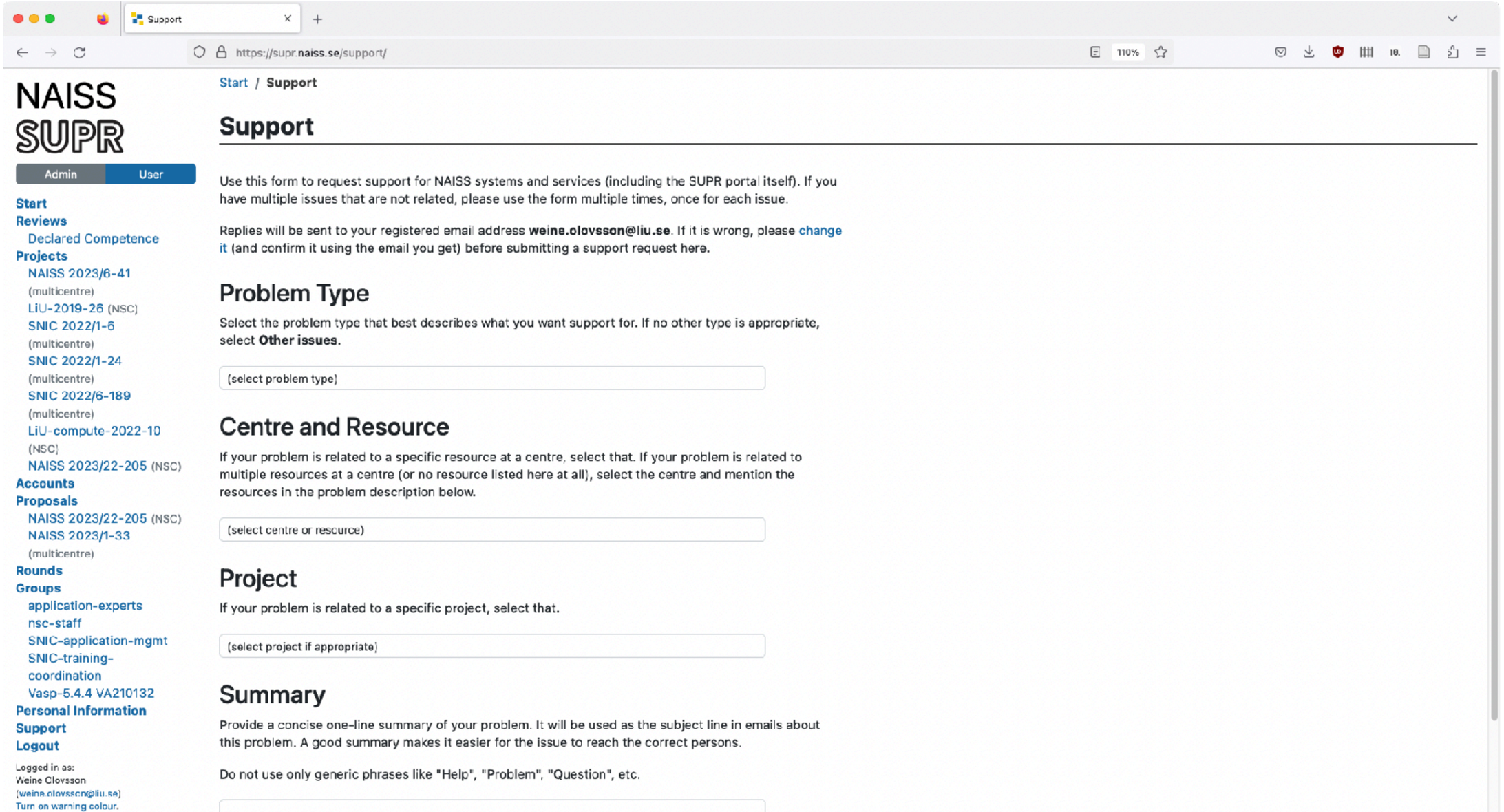
Preferred entry for support questions!

https://supr.naiss.se/

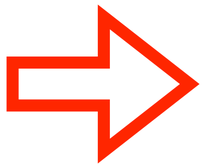
Support via SUPR

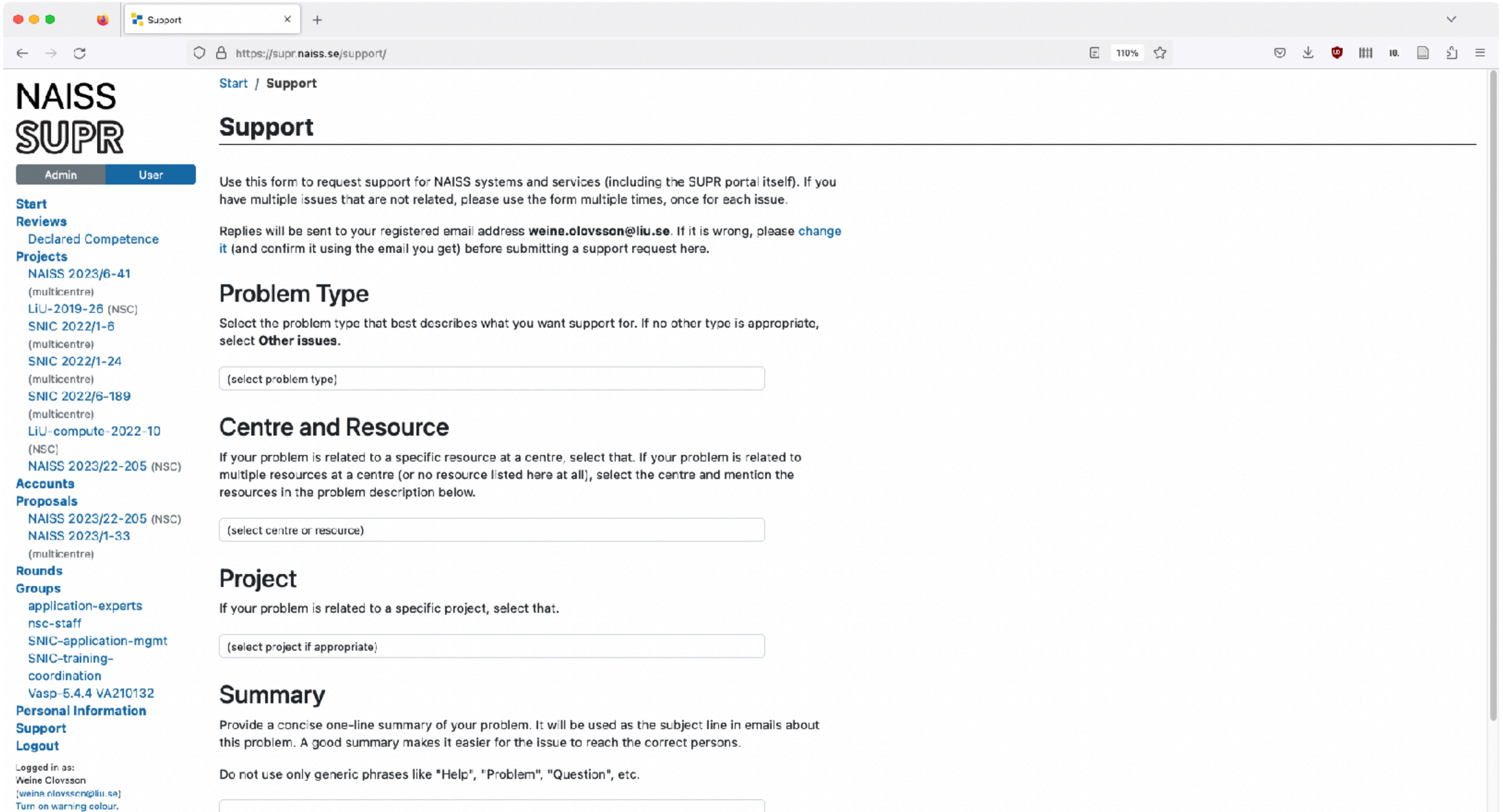


Support via SUPR

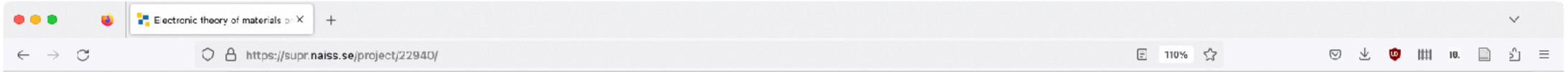


Projects in SUPR





Projects in SUPR



The Principal Investigator and the proxy can add and remove members in SUPR. Membership in this project can also be requested using the Request Membership in Project function. The Principal Investigator and the proxy will then get an email with a link in it, that is used to approve or deny the membership request.

Leave Project

Leave Project

Storage projects linked to this compute project

Members of this compute project become extended members of the linked storage project and can access its storage.

Storage Project	Title	PI
SNIC 2022/6-10	Storage for theoretical physics environm...	Rickard Armiento
NAISS 2023/6-41	Storage for theoretical physics environm...	Rickard Armiento

Resources

Allocation shows the current allocation.

Compute

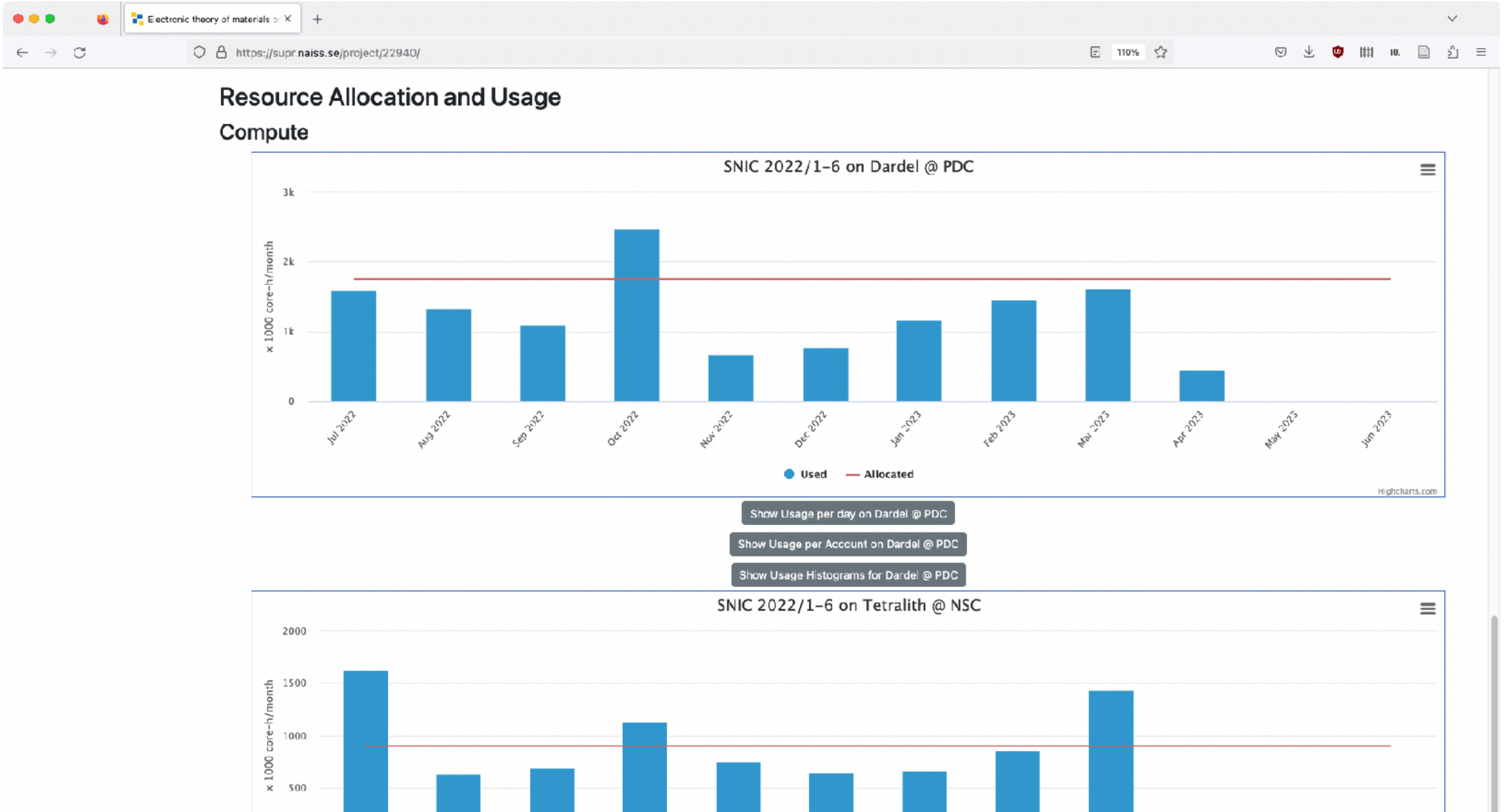
Total Allocation during the whole project is shown with a **Percentage** field to the right, that compares **Total Usage** with the total allocation. The **Allocation until Today** field shows the allocation until today, also with a **Percentage** comparison.

Resource	Allocation	Unit	Total Usage	Allocation until Today	Percentage	Total Allocation	Percentage
Dardel @ PDC	1750	x 1000 core-h/month	12646.8	16450.0	76.9 %	21000.0	60.2 %
Tetralith @ NSC	900	x 1000 core-h/month	8652.5	8460.0	102.3 %	10800.0	80.1 %

Resource Allocation and Usage

Compute

Projects in SUPR



When & Why to use HPC?

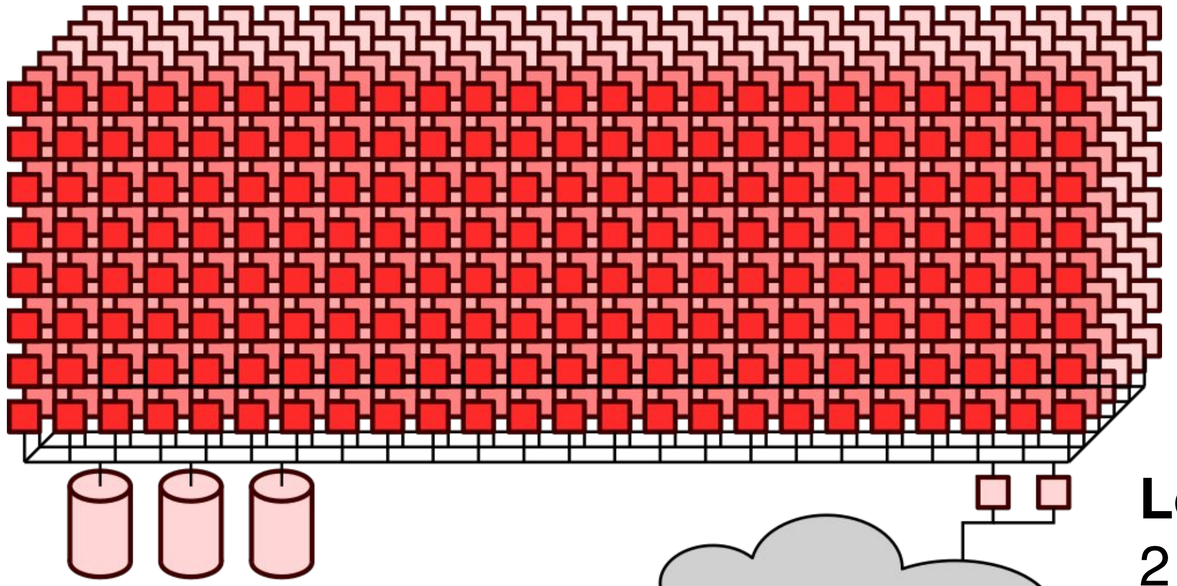
HPC = High Performance Computing

- High number of simulation or data analysis jobs
- The jobs are too large for a desktop/laptop
- Used in most research fields today
  - Numerical weather prediction
  - Climate simulations
  - Flow simulations
  - Materials science
  - Many disciplines within Chemistry, Physics, Biology
  - …

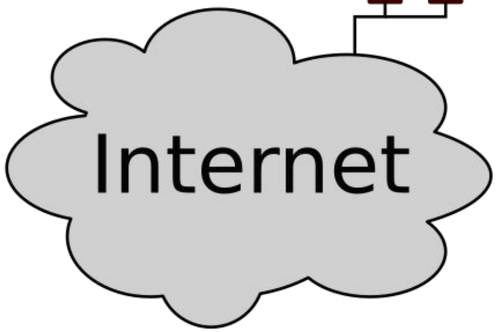
Desktop PC vs HPC

Desktop/laptop:
8 cores
16 GB RAM
Windows, MacOS (Unix), Linux
1 user

Tetralith: 1908 nodes, **Sigma:** 110 nodes



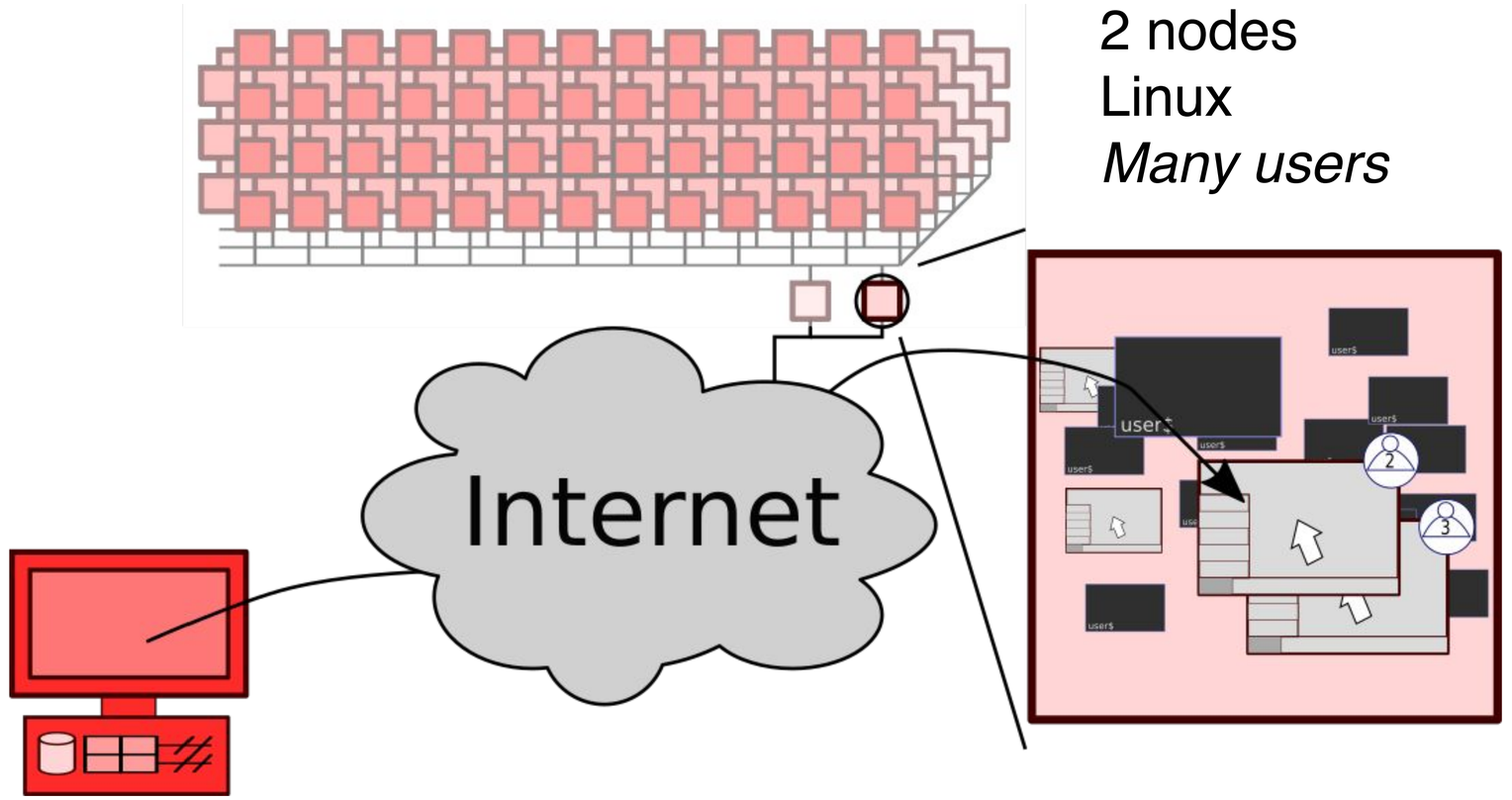
Work node:
32 cores
96 (384) GB RAM
Linux
Omni-Path network
1 - few users at a time



Login nodes:
2 nodes
Linux
Many users

Access to Tetralith

- Typical access: using ssh
- For graphics, use ThinLinc
- Many users share login node
- Be mindful of login node usage
- Work node access via queue system (Slurm)



Login nodes:
2 nodes
Linux
Many users

https://www.nsc.liu.se/support/getting-started/

Access to Tetralith: ssh

ssh: the common, classical way, to login

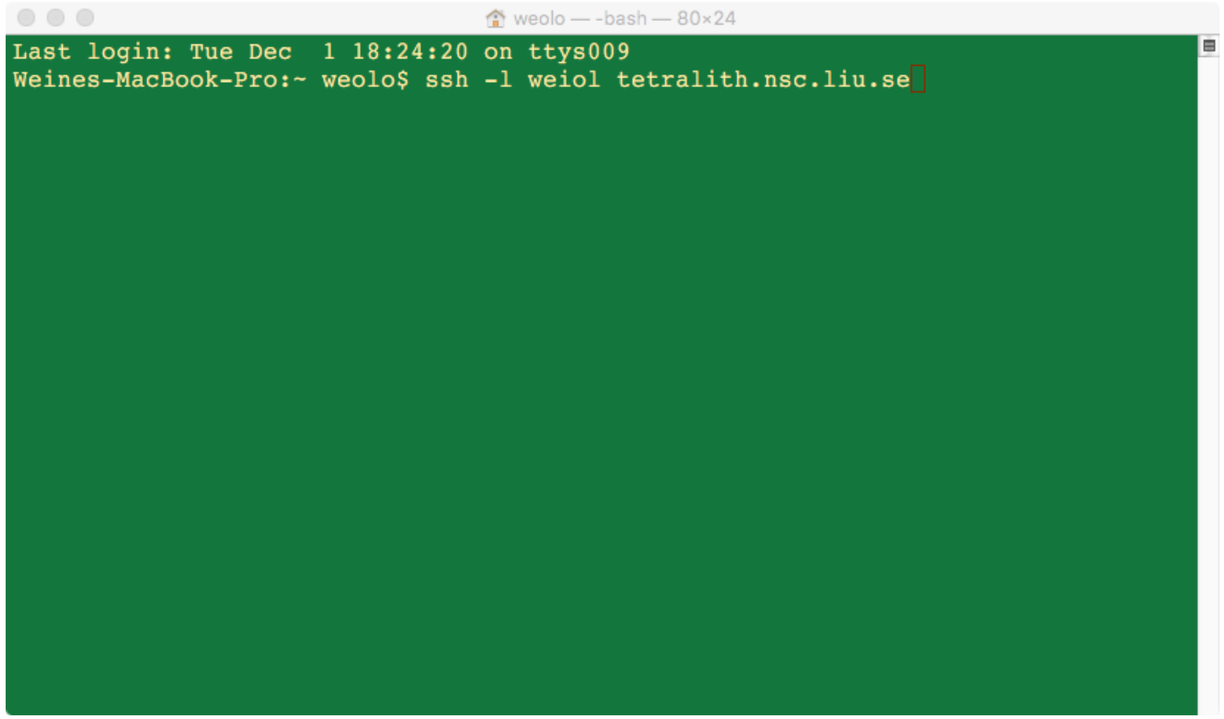
Typical login via terminal from Linux / Mac:

```
ssh username@tetralith.nsc.liu.se
```

- Windows: can use PuTTY

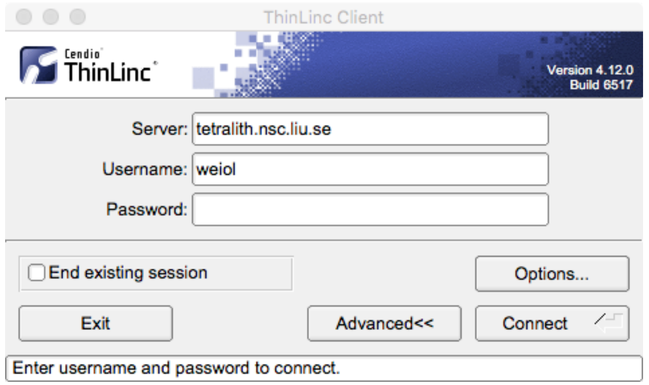
Note: to end up on a specific login node use:

```
tetralith1.nsc.liu.se
tetralith2.nsc.liu.se
```

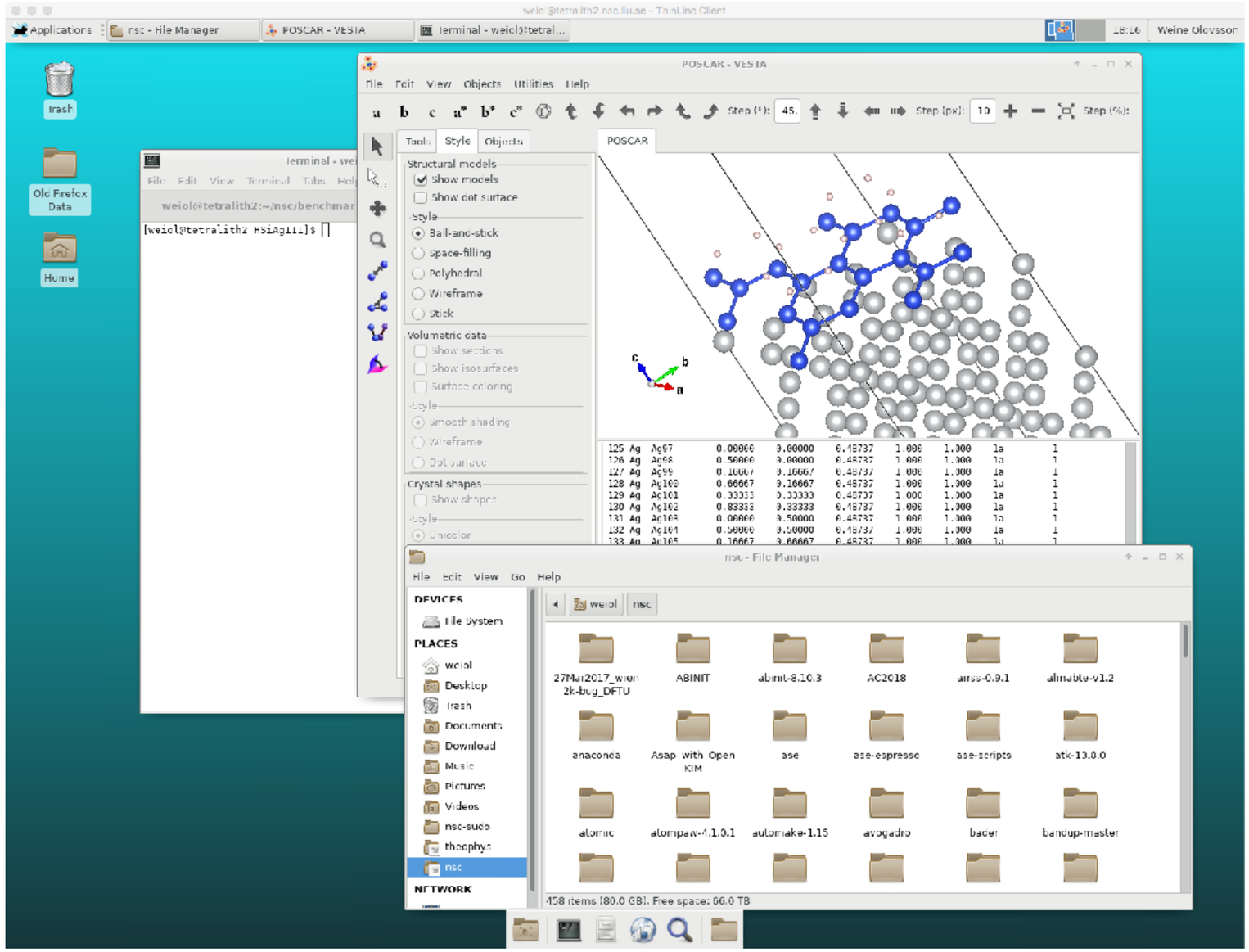


https://www.nsc.liu.se/support/getting-started/

Access to Tetralith: ThinLinc



- ThinLinc - *virtual desktop*
- Persistent sessions (compare screen, tmux)
- Recommended for graphics
- Hardware acc. graphics (vglrun) in some cases



https://www.nsc.liu.se/support/graphics/

Some Basics

- **Linux**, see e.g. guide and forum
  - Basic commands: `cd, pwd, ls, mkdir, mv, grep, less, cat, …`
- Common tools
  - Text editors: `vi, gedit, emacs, nano, …`
  - Plotting graphs: `gnuplot, grace, …`
  - Analysis (basic/complex): `python, R, Matlab, …`
- Useful things
  - Persistent terminal session: `screen, tmux`
  - Check compute usage: `projinfo`
  - Check disk usage: `nscquota`

Files & Storage

Three types of storage areas available:	Recover deleted files? Backup?	Snapshot?
1. Personal home directory, e.g. `/home/x_user`	yes!	yes!
2. Project storage, owned by PI, e.g. `/proj/ourstuff`	no!	yes!
3. Work node local disk (during runs)	no!	no!

Some notes:

- Use `nscquota` to check available disk space
- Project storage is linked to specific project allocation and life time
- Good idea to have your own backup
- Data is never 100% safe, there's always some risk

https://www.nsc.liu.se/support/storage/snic-centrestorage/recover-deleted-files/

https://www.nsc.liu.se/support/storage/index.html

Basic Security

- Unique password (non-trivial but not overly complicated)
- Suspicion that your account is compromised -> contact NSC

  - Don’t hesitate to contact us!
- Sharing accounts is not allowed (accounts are personal)

  Share files e.g. by managing project memberships and use /proj

https://www.nsc.liu.se/support/security/ Also, on how to use key-pair authentication

Software: How do I get Code X?

1. Check installed software webpage
2. Check module system (`module avail`)
3. Ask NSC support
4. Build and install yourself

Special wrappers/rec. e.g.: Gaussian

NSC software installation policy:

- Users encouraged to install in /home or /proj
- NSC can help to install on request

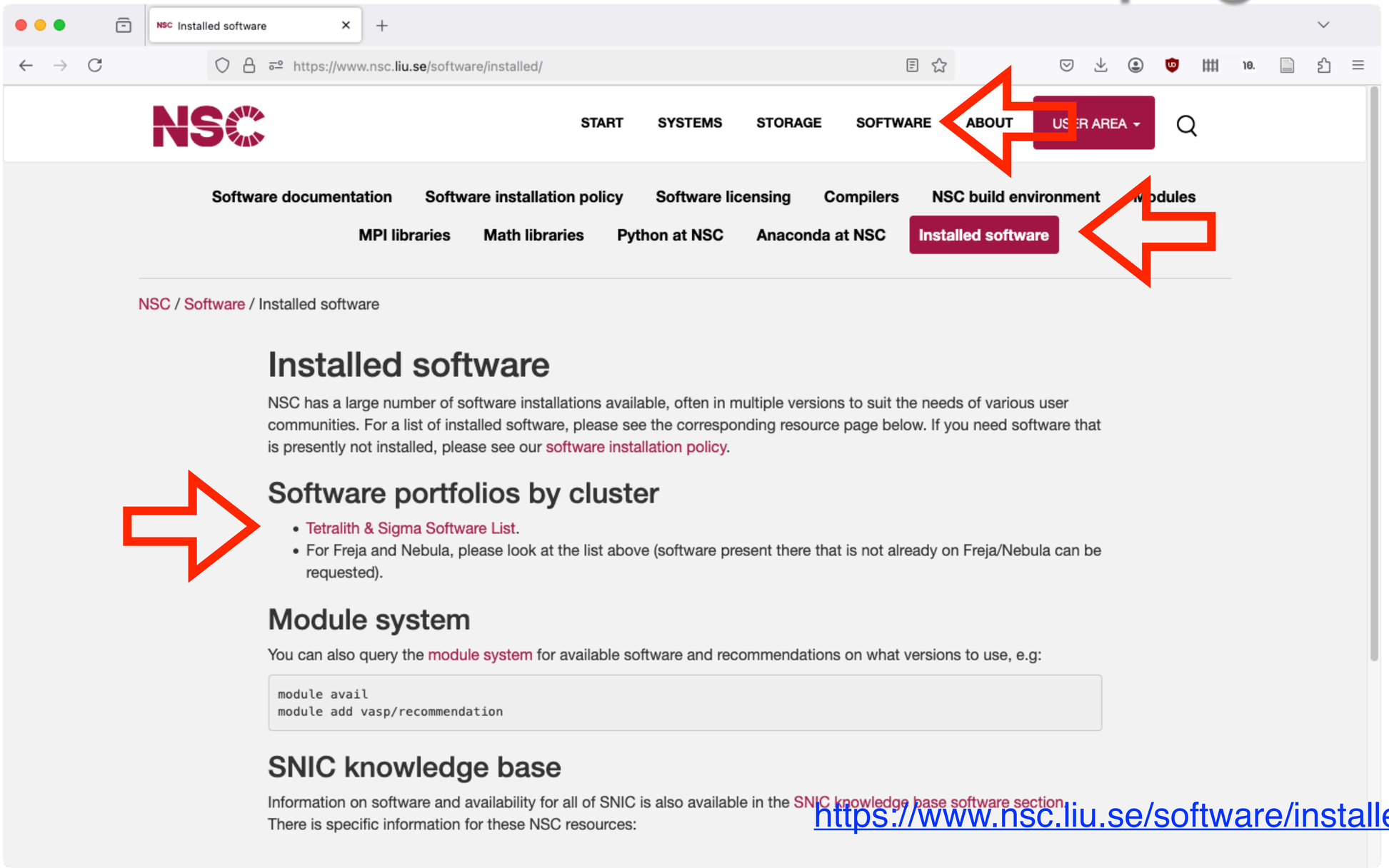
Testing, benchmarking, optimization

Global installation: wide or not usage, license?

https://www.nsc.liu.se/software/installed/

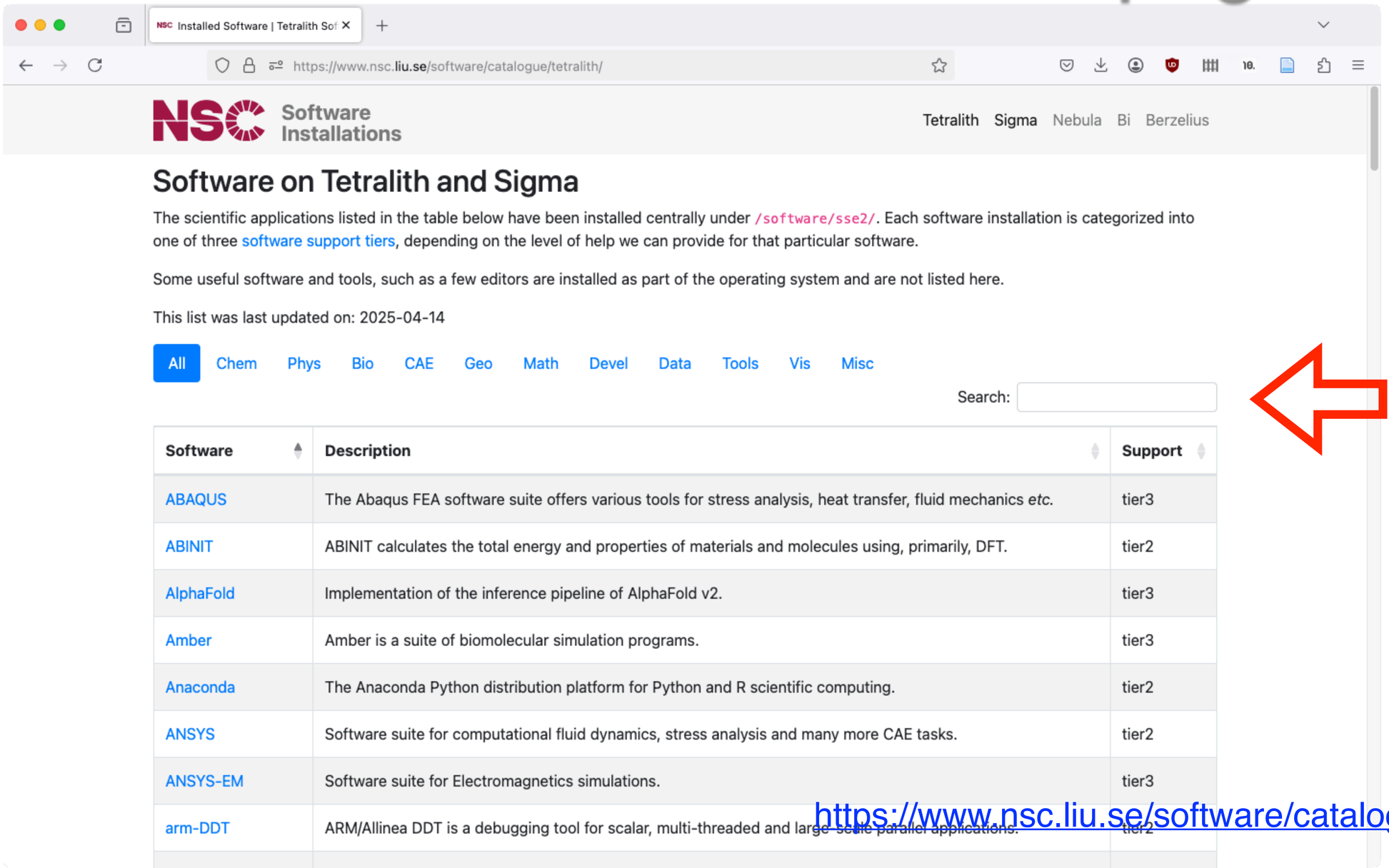
https://www.nsc.liu.se/software/installation-policy/

Software: Installation Webpage



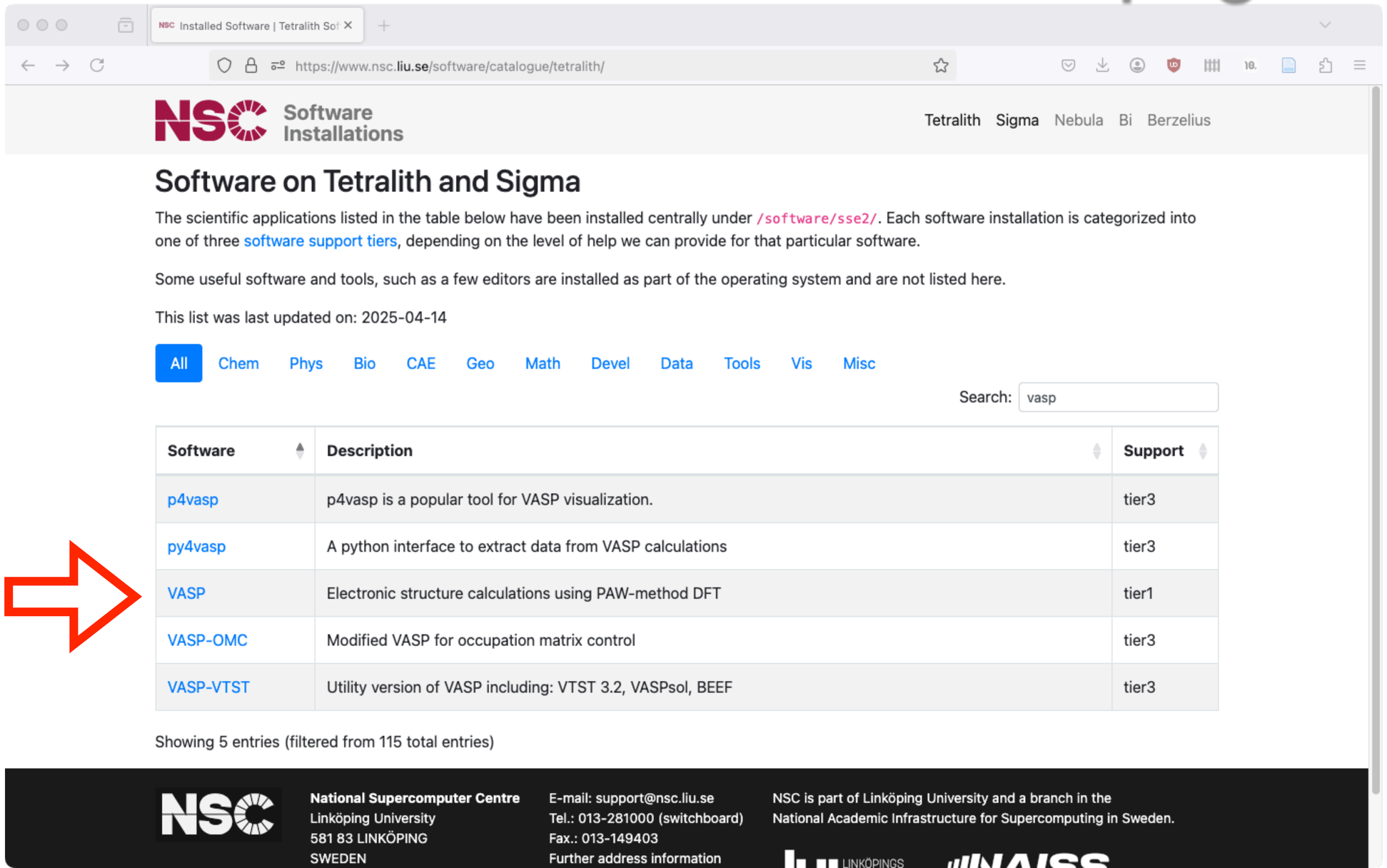
https://www.nsc.liu.se/software/installed/

Software: Installation Webpage

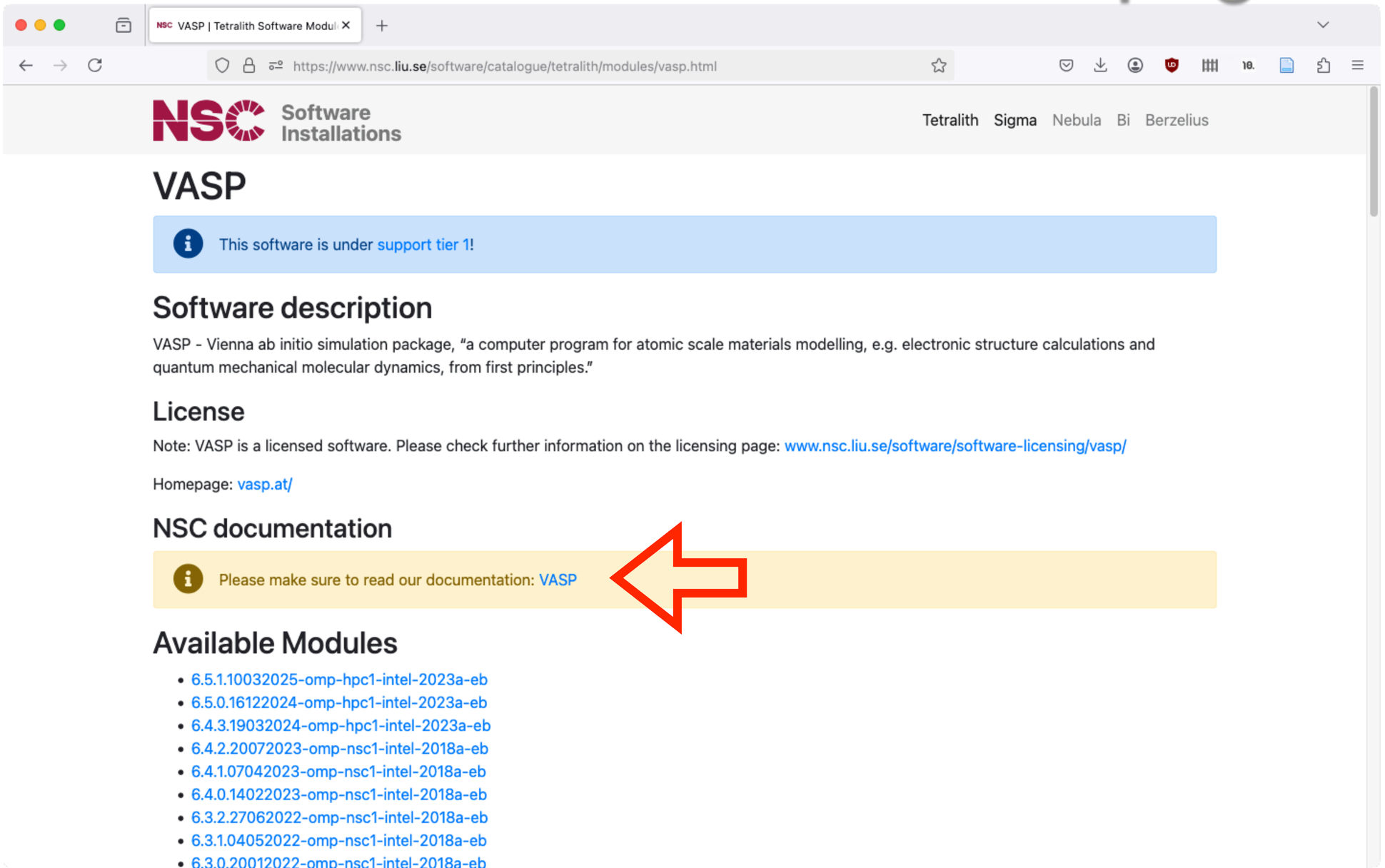


https://www.nsc.liu.se/software/catalogue/tetralith/

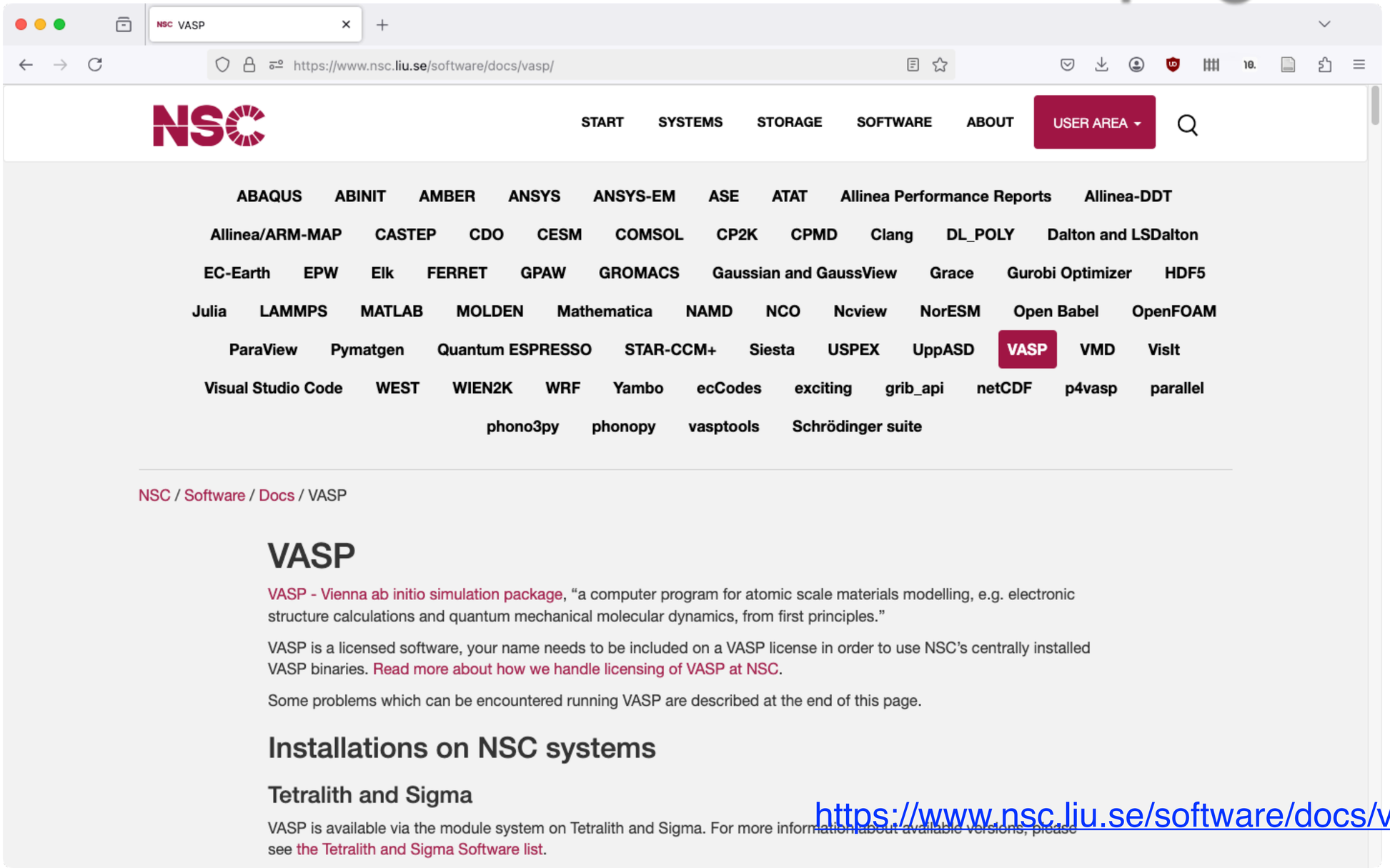
Software: Installation Webpage



Software: Installation Webpage



Software: Installation Webpage



https://www.nsc.liu.se/software/docs/vasp/

Software: Module System

`module help …`	Show information for module …
`module avail`	List available modules
`module avail …`	Search after module containing … in its name
`module add …`	Add a module (same as module load …)
`module list`	List your loaded modules
`module rm …`	Remove the … module
`module purge`	Remove all loaded modules (useful to start “clean”)

https://www.nsc.liu.se/software/modules/

Software: Module System

NSC module usage:

- Only load specific software module (not dependencies)

  at many other centers, must load all dependencies

- Only load build environment when building

  gives access to specific build time modules

https://www.nsc.liu.se/software/modules/

Software: Module System

```
[weiol@tetralith2 ~]$ module avail vasp

------------------------------------------------------ /software/sse/modules-------------------------------------------------------
   p4vasp/recommendation                                    (D)    VASP-VTST/3.2-sol-5.4.4.16052018-vanilla-nsc1-intel-2018a-eb
   VASP/5.4.4.16052018-wannier90-nsc1-intel-2018a-eb               VASP/6.3.0.20012022-omp-nsc1-intel-2018a-eb
   p4vasp/tmp1                                                     VASP-VTST/4.2-6.3.0.20012022-nsc1-intel-2018a-eb
   VASP/5.4.4.16052018-wannier90-nsc2-intel-2018a-eb               VASP/6.3.1.04052022-omp-nsc1-intel-2018a-eb
   p4vasp/0.3.30-nsc1                                              VASP/recommendation                                          (D)
   VASP/6.1.0.28012020-nsc1-intel-2018a-eb                         VASP/6.3.2.27062022-omp-nsc1-intel-2018a-eb
   VASP-OMC/recommendation                                  (D)    VASP/5.4.4.16052018-nsc1-intel-2018a-eb
   VASP/6.1.2.25082020-nsc1-intel-2018a-eb                         vasptools/0.3
   VASP-OMC/5.4.4.16052018-nsc1-intel-2018a-eb                     VASP/5.4.4.16052018-nsc1-intel-2018b-eb
   VASP/6.1.2.25082020-omp-nsc1-intel-2018a-eb.                    VASP-VTST/recommendation                                (D)
   VASP/5.4.4.16052018-nsc2-intel-2018a-eb                         VASP/6.2.0.14012021-omp-nsc1-intel-2018a-eb
   VASP-VTST/3.2-sol-5.4.4.16052018-nsc2-intel-2018a-eb            VASP/5.4.4.16052018-vanilla-nsc1-intel-2018a-eb
   VASP/6.2.1.29042021-omp-nsc1-intel-2018a-eb

  Where:
   D:  Default Module

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


[weiol@tetralith2 ~]$
```

https://www.nsc.liu.se/software/modules/

Software: Compilers and Libraries

NSC recommendation: to compile your own software, load a build environment

- Compilers
  - Intel: icc, ifort
  - Gcc: gcc, gfortran
- MPI libraries
  - Intel (impi), OpenMPI
- Math libraries
  - e.g. MKL
- Build environments
  - e.g. `buildenv-intel/2023a-eb`

https://www.nsc.liu.se/software/index.html

Software: Build Environment

```
[weiol@tetralith1 ~]$ module avail buildenv

------------------------------------------ /software/sse/modules-------------------------------------------------
   buildenv-gcc/recommendation (D)                buildenv-gcc/2016b-eb                buildenv-gcccuda/11.4-9.3.0-bare
   buildenv-intel/2015.1.133-impi-2018.1.163-eb   buildenv-intel/2018b-eb              buildenv-nvhpc/recommendation (D)
   buildenv-gcc/7.3.0-bare                        buildenv-gcc/2018a-eb                buildenv-impi-gcc/recommendation (D)
   buildenv-intel/2016b-eb                        buildenv-intel/2018.u1-bare          buildenv-gcc/9.3.0-bare
   buildenv-gcccuda/recommendation  (D)           buildenv-impi-gcc/2018a-eb           buildenv-intel/2017.u7-bare
   buildenv-intel/2021.3.0-oneapi                 buildenv-gcc/11.3.0-bare             buildenv-gcccuda/10.2-7.3.0-bare
   buildenv-intel/recommendation    (D)           buildenv-intel/2018a-eb              buildenv-nvhpc/cu11.4-22.1-bare

  Where:
   D:  Default Module

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


[weiol@tetralith1 ~]$
```

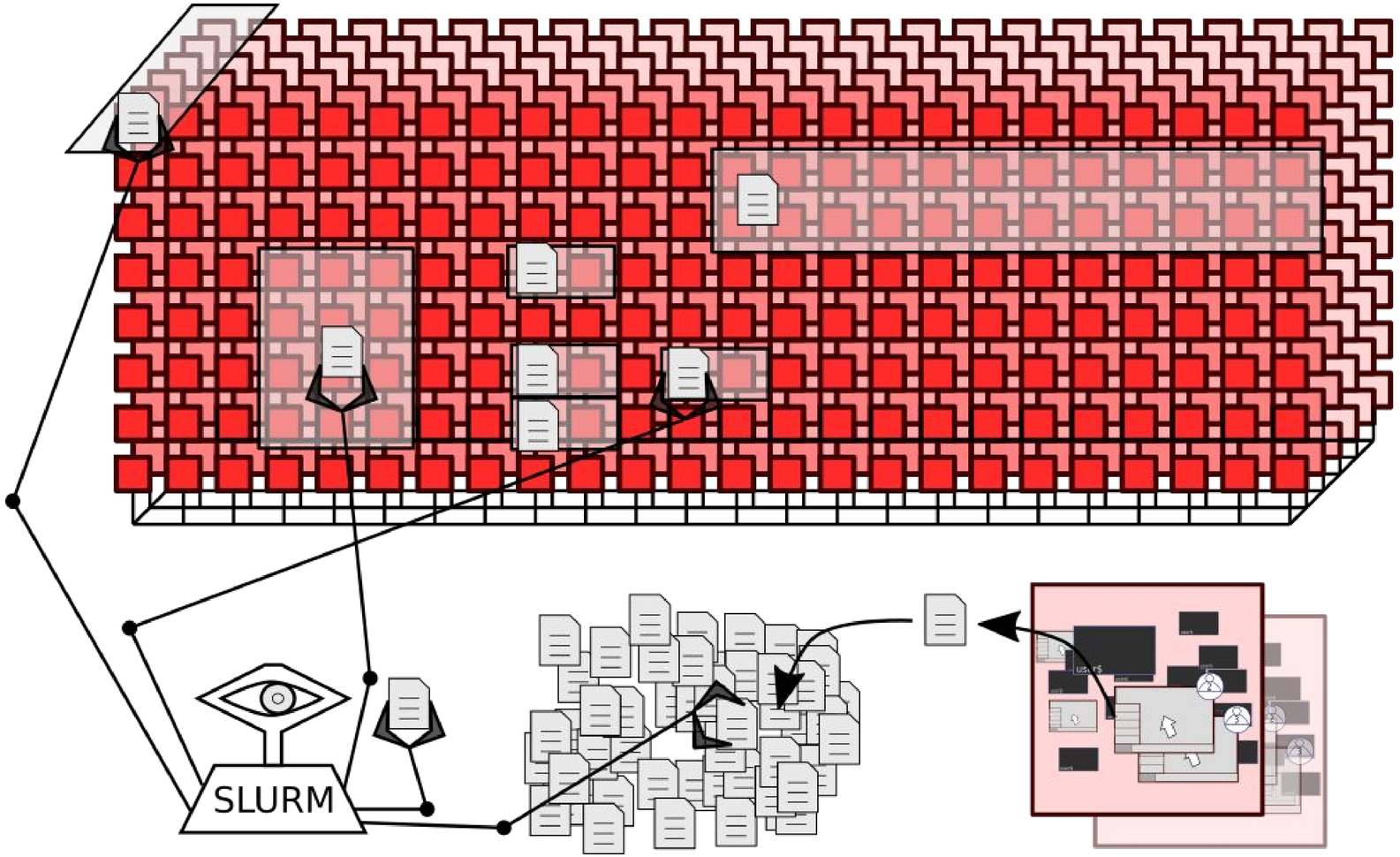
https://www.nsc.liu.se/software/modules/

Software: Build Environment

```
[weiol@tetralith1 ~]$ module add buildenv-intel/2018a-eb
**************************************************
You have loaded an intel buildenv module
**************************************************
The buldenv-intel module makes available:
 - Compilers: icc, ifort, etc.
 - Mpi library with mpi-wrapped compilers: intel mpi with mpiicc, mpiifort, etc.
 - Numerical libraries: intel MKL

It also makes a set of dependency library modules available via
the regular module command. Just do:
  module avail
to see what is available.

NOTE: You shoud never load build environments inside submitted jobs.
(with the single exception of when using supercomputer time to compile code.)

[weiol@tetralith1 ~]$ module list

Currently Loaded Modules:
  1) mpprun/4.1.3                          5) buildtool-easybuild/4.3.0-nscde3532a       9) ifort/.2018.1.163-GCC-6.4.0-2.28 (H)  13)
buildenv-intel/2018a-eb
  2) nsc/.1.1                     (H,S)    6) GCCcore/6.4.0                             10) impi/.2018.1.163                 (H)
  3) EasyBuild/4.3.0-nscde3532a            7) binutils/.2.28                      (H)  11) imkl/.2018.1.163                 (H)
  4) nsc-eb-scripts/1.2                    8) icc/.2018.1.163-GCC-6.4.0-2.28      (H)  12) intel/2018a

  Where:
   S:  Module is Sticky, requires --force to unload or purge
   H:             Hidden Module



[weiol@tetralith1 ~]$
```

Queue System: Slurm

- Many jobs & users
- Resource access via Slurm
- Several methods:
  - `sbatch`
  - `interactive`
- Run as much possible, based on prior usage
- Fairshare scheduling with backfill
- 168 hours (7d) walltime limit
- Avoid short time wide jobs, “flat jobs”
- Priority boosting available



https://www.nsc.liu.se/support/batch-jobs/tetralith/

Slurm: Running Batch Job

- Regular production runs
- Output to files

  `slurm-JOBID.out`

Example: a job script called “run.sh”

```
#!/bin/bash
#SBATCH -A naiss2024-00-00
#SBATCH -t 1:00:00
#SBATCH -n 32
#SBATCH -J vasptst

module load VASP/6.4.3.19032024-omp-hpc1-intel-2023a-eb
mpprun vasp_std
```

project
time
MPI ranks
job name

NSC MPI job launching tool

Submit job:
`sbatch run.sh`

Check queue:
`squeue -u USERNAME`

Checking jobs:
`jobload JOBID`
`jobsh NODE` login to node, run “top”
`seff JOBID`
`lastjobs`

https://www.nsc.liu.se/support/batch-jobs/introduction/

Slurm: Interactive Job

- Testing, debugging
- Hands-on, direct node access

  Example: similar settings as for the job script

```
[weiol@tetralith1 ~]$interactive -A naiss2020-13-76 -n 32 -t 1:00:00
salloc: Pending job allocation 11193334
salloc: job 11193334 queued and waiting for resources
salloc: job 11193334 has been allocated resources
salloc: Granted job allocation 11193334
srun: Step created for job 11193334
[weiol@n405 ~]$
```

- Special queue for brief testing, max 1h, max 1 node (also with job script)

  `--reservation=now` @Tetralith

  `--reservation=devel` @Sigma

https://www.nsc.liu.se/support/batch-jobs/tetralith/

mpprun (update)

- **NSC** MPI job launching tool
- Extra features available, check with:

```
[weiol@tetralith1 ~]$mpprun --help
usage: mpprun [-h] [--version] [-n NRANKS] [--launcher LAUNCHER] [--handler HANDLER] [--no-smt] [-
C] [--pass EXTRA_LAUNCH_ARGS] [--no-modules] [-c CPUS_PER_TASK] [-d] [-v] [-q] [-i] [--allinfo]
executable ...

This is a helper program to figure out what MPI launcher to use, with which arguments, in what
environment (including which HPC modules to load) when launching a binary or script that uses MPI.

positional arguments:
  executable            The binary or script to execute.
  arguments             Arguments to pass to the executable.

...
```

+ more lines of detailed description

https://www.nsc.liu.se/software/buildenv/

Tetralith/Sigma OS upgrade

- CentOS 7 (El7) -> Rocky Linux 9 (**El9**) ← Security updates
- Disk space (`/home`, `/proj`) not affected
- Dec'23 - Jan'24 (finished), mpprun was updated
- Might need to modify scripts, use different modules, reinstall software etc.
- Modules missing, software not working, other questions?

→ https://supr.naiss.se/support

https://www.nsc.liu.se/support/systems/tetralith-os-upgrade/

https://www.nsc.liu.se/support/systems/sigma-os-upgrade/

Best Practices & Suggestions

In general:

- Be careful how you use Tetralith/Sigma login nodes
- Use SUPR to follow project usage
- Use the NSC documentation

contact us if problems!
we try to describe everything…

- Be careful about what you put in .bashrc (keep as simple as possible)
- Don’t hesitate to contact us for help/questions!

https://supr.naiss.se/support support@nsc.liu.se

Best Practices & Suggestions

Common problems:

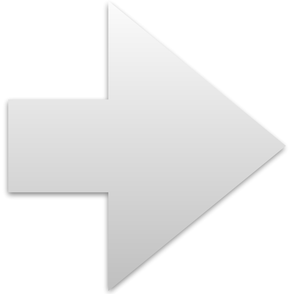
- My job **failed/crashed**. What now?
  - First, try to understand the cause
  - Fill in support form https://supr.naiss.se/support

    provide details! username, system, jobid, job path, …
- Odd problems (lots of things set in .bashrc?)
- Don’t run heavy stuff / production work on the login node
  - For brief testing e.g. run interactively `--reservation=now` @Tetralith `--reservation=devel` @Sigma

https://www.nsc.liu.se/support/getting-help/

Further Resources

- Working effectively on Tetralith / Sigma 2018
- Working with Python on Tetralith 2019
- NSC introduction day 2017 More details, e.g. running calcs.



Check links for presentations (.pdf)

- Presentations available at webpage!

- Working effectively with HPC systems