Practical Examples

National Supercomputer Centre (NSC), Linköping University
SNIC-PRACE training
Online @NSC 20th Apr 2021, 10:00 - ca. 15:00
Before Running

• Software already available at center?
  ```
  module avail name
  ```

• Testing before production runs!
  ```
  Does it work as expected?
  ```

• Reasonable resource usage?
  ```
  projinfo, snicquota
  ```

• Don’t hesitate to ask us questions!

  support@nsc.liu.se
During and after Run

• `jobload $JOBID` - [NSC] resource usage running job

• `seff $JOBID` - summary on a finished job

• `lastjobs` - [NSC] list your 10 last jobs

`squeue -u $USERNAME $JOBID, $NODE`
At the Work Node

- `jobsh $NODE` - [NSC] first, login (instead of `ssh`)
- `top` and `ps` - check running processes
- `htop` - info on e.g. threads
- `hwloc-ps` - check bound processes
- `collectl` - monitoring tool

`squeue -u $USERNAME $JOBID, $NODE`
Example

```bash
$ cat run.sh
#!/bin/bash
#SBATCH -A nsc
#SBATCH -N 4
#SBATCH --ntasks-per-node=4
#SBATCH -t 6:00:00
#SBATCH -J vasp_n4omp
export OMP_NUM_THREADS=6
export KMP_STACKSIZE=256m
module load VASP/6.1.2.25082020-omp-nsc1-intel-2018a-eb
mpirun vasp_gam

$ sbatch run.sh
Submitted batch job 12914354

$ squeue -u weiol

$ jobload 12914354
```

4 nodes reserved
4 tasks (MPI-ranks)/node
6 OpenMP threads/node

4 ranks x 6 threads = 24
1 core = 100%, expects

cpu%: 2400
Example

```bash
$ jobsh n606
[weiol@n606 ~]$ htop
```

![htop output](image-url)
Example

```bash
$ seff 12914354
 Job ID: 12914354
 Cluster: tetralith
 User/Group: weiol/weiol
 State: TIMEOUT (exit code 0)
 Nodes: 4
 Cores per node: 32
 CPU Utilized: 23-04:35:03
 CPU Efficiency: 72.38% of 32-00:59:44 core-walltime
 Job Wall-clock time: 06:00:28
 Memory Utilized: 301.87 GB (estimated maximum)
 Memory Efficiency: 83.16% of 363.00 GB (2.84 GB/core)
```

TIMEOUT at 06:00:28 h, since

#SBATCH -t 6:00:00
Fe 2000 atoms, PBE, MD
Memory heavy!

MPI:
4 ranks / node

OpenMP (hybrid):
4 ranks x 4 threads / node

-C fat:
Select “fat” memory nodes
384 vs 96 GiB RAM
increase ranks!
Best Practices & Suggestions @NSC

In general:

- Be careful how you use Tetralith/Sigma login nodes
- Use SUPR to follow project usage
- Use the NSC documentation
- Be careful about what you put in .bashrc (keep as simple as possible)
- Don’t hesitate to contact support@nsc.liu.se for help/questions

https://www.nsc.liu.se/support/getting-help/
Best Practices & Suggestions @NSC

Common problems:

- My job **failed/crashed**. What now?
  - First, try to understand the cause
  - Contact **support@nsc.liu.se** / fill in form [https://supr.snic.se](https://supr.snic.se)
    - **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=devel

[https://www.nsc.liu.se/support/getting-help/](https://www.nsc.liu.se/support/getting-help/)