

Relion Prometheus usage guide

- [Interactive Relion job with Relion GUI](#)
- [Relion in SLURM batch jobs](#)
- [Submitting SLURM jobs from Relion GUI](#)

Relion could be used at [Prometheus](#) supercomputer in three ways:

- inside graphical interactive job using `pro-viz` service (Main documentation - polish - [Obliczenia w trybie graficznym: pro-viz](#))
- in SLURM batch job though SLURM script submitted from command line
- in SLURM batch job submitted from Relion GUI started via `pro-viz` service in dedicated partition

Interactive Relion job with Relion GUI

In order to start interactive Relion job with access to Relion GUI

1. Log into Prometheus login node

Log into Prometheus login node

```
ssh <login>@pro.cyfronet.pl
```

2. Load `pro-viz` module

Load pro-viz module

```
module load tools/pro-viz
```

3. Start `pro-viz` job
 - a. Submit `pro-viz` job to queue
 - i. CPU-only job

Submission of CPU pro-viz job

```
pro-viz start -N <number-of-nodes> -P <cores-per-node> -p <partition/queue> -t <maximal-time> -m <memory>
```

- ii. GPU job

Submission of GPU pro-viz job

```
pro-viz start -N <number-of-nodes> -P <cores-per-node> -g <number-of-gpus-per-node> -p <partition/queue> -t <maximal-time> -m <memory>
```

- b. Check status of submitted job

Status of pro-viz job(s)

```
pro-viz list
```

- c. Get password to `pro-viz` session (when job is already running)\

Pro-viz job password

```
pro-viz password <JobID>
```

exemple output

Pro-viz password example output

Web Access link:

`https://viz.pro.cyfronet.pl/go?c=<hash>&token=<token>`

link is valid until: Sun Nov 14 02:04:02 CET 2021

session password (for external client): <password>

full commandline (for external client): `vncviewer -SecurityTypes=VNC,UnixLogin,None -via <username>@pro.cyfronet.pl -password=<password> <worker-node>:<display>`

d. Connect to graphical pro-viz session

i. you could use weblink obtained in previous point

ii. you could use VNC client (i.e. TurboVNC). Configuration of client described in [Obliczenia w trybie graficznym: pro-viz](#) (in polish)

4. Setup Relion environment

a. When connected to GUI open Terminal and load Relion module

Load Relion module

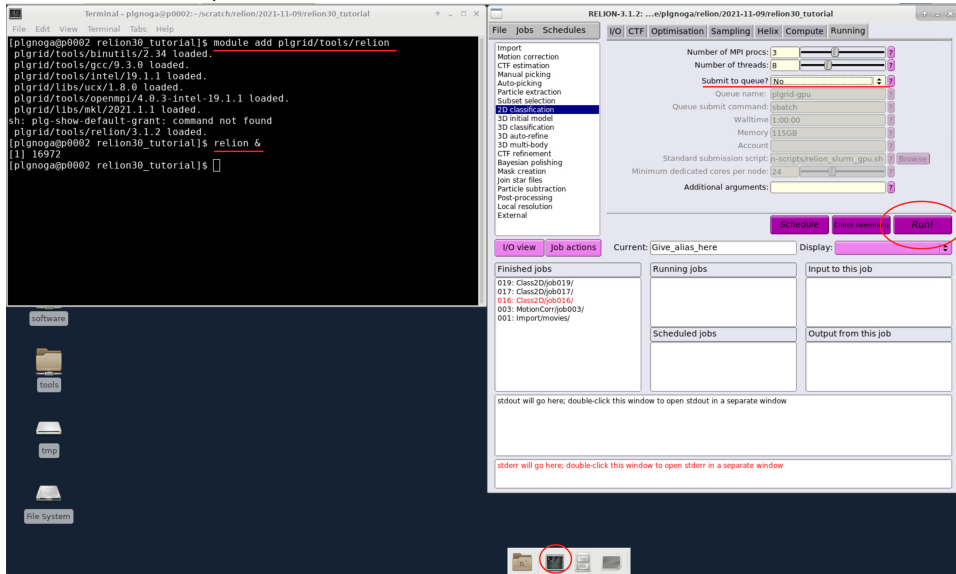
```
module load plgrid/tools/relion
```

b. Start Relion GUI in background

Start relion

```
relion &
```

5. Use Relion GUI for computation.



6. After finishing work terminate job

Pro-viz job password

```
pro-viz stop <JobID>
```

Relion in SLURM batch jobs

Most of Relion jobs could be run as batch jobs using SLURM

1. Log into Prometheus login node

Log into Prometheus login node

```
ssh <login>@pro.cyfronet.pl
```

2. Move to Relion project directory

Change directories

```
cd $SCRATCH/<relion-project>
```



Usage of filesystems

Relion project during computations should be stored in \$SCRATCH filesystem on Prometheus. More info - https://kdm.cyfronet.pl/portal/Prometheus:Basics#Disk_storage. For longer storage user should use \$PLG_GROUPS_STORAGE/<team_name> filesystem.

3. Submit job

Job submission

```
sbatch script.slurm
```

a. Example CPU-only SLURM script

Relion CPU-only SLURM script

```
#!/bin/bash
# Number of allocated nodes
#SBATCH --nodes=1
# Number of MPI processes per node
#SBATCH --ntasks-per-node=4
# Number of threads per MPI process
#SBATCH --cpus-per-task=6
# Partition
#SBATCH --partition=plgrid
# Requested maximal walltime
#SBATCH --time=0-1
# Requested memory per node
#SBATCH --mem=110GB
# Computational grant
#SBATCH --account=<name-of-grant>

export RELION_SCRATCH_DIR=$SCRATCHDIR

module load plgrid/tools/relion/3.1.2
mpirun <relion-command>
```

b. Example GPU SLURM script

Relion GPU SLURM script

```
#!/bin/bash
# Number of allocated nodes
#SBATCH --nodes=1
# Number of MPI processes per node
#SBATCH --ntasks-per-node=4
# Number of threads per MPI process
#SBATCH --cpus-per-task=6
# Partition
#SBATCH --partition=plgrid-gpu
# Number of GPUs per node
#SBATCH --gres=gpu:2
# Requested maximal walltime
#SBATCH --time=0-1
# Requested memory per node
#SBATCH --mem=110GB
# Computational grant
#SBATCH --account=<name-of-grant>

export RELION_SCRATCH_DIR=$SCRATCHDIR

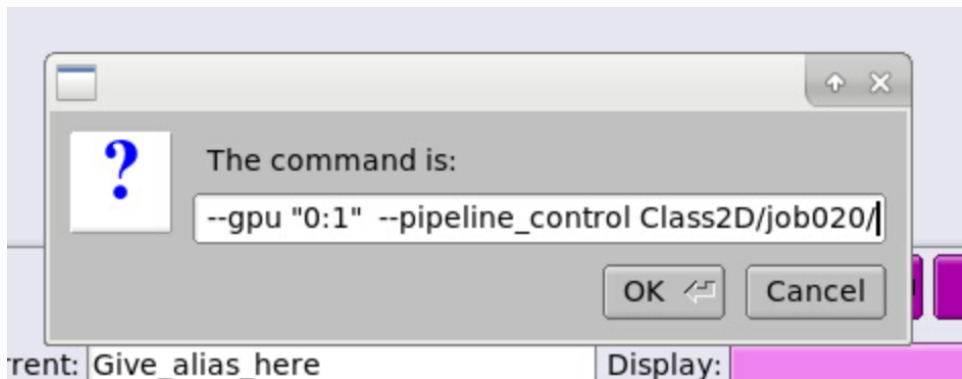
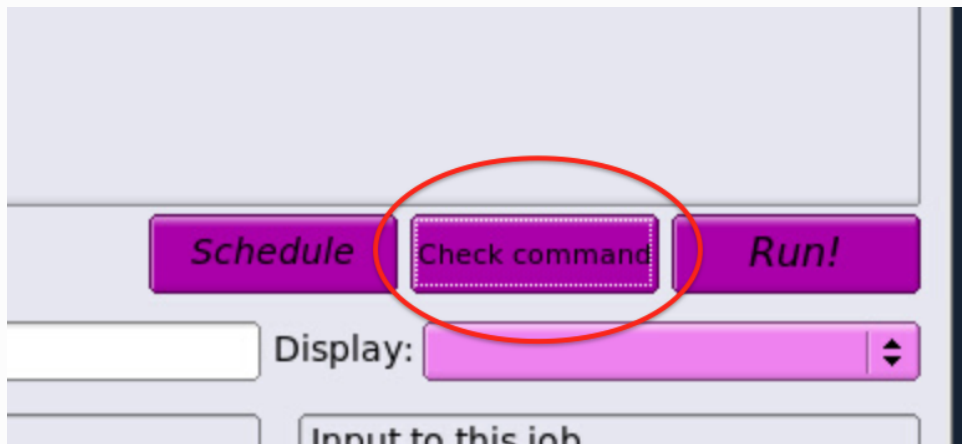
module load plgrid/tools/relion/3.1.2
mpirun <relion-command> --gpu $CUDA_VISIBLE_DEVICES
```

GPUs usage

GPUs are available only for selected grants in partitions `plgrid-gpu` and `plgrid-gpu-v100`. One should always use `--gpu $CUDA_VISIBLE_DEVICES` to request GPUs allocated for job.

Relion command

Relion command syntax could be checked using GUI and copied to script



Job submission
squeue

or

Job submission
pro-jobs

Submitting SLURM jobs from Relion GUI

1. Start job as in [pro-viz session](#) but using `plgrid-services` partition/queue.
2. In Relion GUI use "Submit to queue" in "Running" tab
 - a. Select submission scripts from directory

The screenshot shows the 'Running' tab of the Relion GUI. The settings are as follows:

- Number of MPI procs: 3
- Number of threads: 8
- Submit to queue? ☒ Yes
- Queue name: plgrid-gpu
- Queue submit command: sbatch
- Walltime: 1:00:00
- Memory: 115GB
- Account:
- Standard submission script: n-scripts/relion_slurm_gpu.sh [Browse](#)
- Minimum dedicated cores per node: 24
- Additional arguments:

At the bottom, the 'Schedule' and 'Run!' buttons are circled in red.

3. Monitor jobs either from Relion GUI or command line using `squeue` or `pro-jobs` commands