Running interactive jobs

Slurm at Prince cluster

Submitting array jobs

Running MPI jobs

Running Matlab jobs

Running Gaussian jobs

Running GPU jobs

Running interactive jobs

Running R jobs

Running Amber jobs

Majority of the jobs on Prince cluster are submitted with the `sbatch` command, and executed in the background. These jobs’ steps and workflows are predefined by users, and their executions are driven by the scheduler system.

There are cases when users need to run applications interactively (interactive jobs). Interactive jobs allow the users to enter commands and data on the command line (or in a graphical interface), providing an experience similar to working on a desktop or laptop. Examples of common interactive tasks are:

- Editing files
- Compiling and debugging code
- Exploring data, to obtain a rough idea of characteristics on the topic
- Getting graphical windows to run visualization
- Running software tools in interactive sessions

Since the login nodes of the Prince cluster are shared between many users, running interactive jobs that require significant computing and IO resources on the login nodes will impact many users.

Interactive jobs on Prince Login nodes

Running compute and IO intensive interactive jobs on the Prince login nodes is not allowed. Jobs may be removed without notice.

Instead of running interactive jobs on Login nodes, users can run interactive jobs on Prince Compute nodes using SLURM's `srun` utility. Running interactive jobs on compute nodes does not impact many users and in addition provides access to resources that are not available on the login nodes, such as interactive access to GPUs, high memory, exclusive access to all the resources of a compute node, etc. **There is no partition on Prince that has been reserved for Interactive jobs.**

Through `srun` SLURM provides rich command line options for users to request resources from the cluster, to allow interactive jobs. Please see some examples and short accompanying explanations in the code block below, which should cover many of the use cases.
In the srun examples below, through "--pty /bin/bash" we request to start bash command shell session in pseudo terminal

# by default the resource allocated is single CPU core and 2GB memory for 1 hour
$ srun --pty /bin/bash

# To request 4 CPU cores, 4 GB memory, and 2 hour running duration
$ srun -c4 -t2:00:00 --mem=4000 --pty /bin/bash

# To request one GPU card, 3 GB memory, and 1.5 hour running duration
$ srun -t1:30:00 --mem=3000 --gres=gpu:1 --pty /bin/bash

In srun there is an option "--x11", which enables X forwarding, so programs using a GUI can be used during an interactive session (provided you have X forwarding to your workstation set up). If necessary please read the wiki pages on how to set up X forwarding for Windows and Linux / Max workstation. NOTE: X forwarding is not required for Slurm tutorial classroom exercises.

# To request computing resources, and export x11 display on allocated node(s)
$ srun --x11 -c4 -t2:00:00 --mem=4000 --pty /bin/bash
$ xterm  # check if xterm popping up okay

# To request GPU card etc, and export x11 display
$ srun --x11 -t1:30:00 --mem=3000 --gres=gpu:1 --pty /bin/bash