

Parallel Computing: Peregrine Instructions

Introduction

Peregrine is the University of Groningen's compute cluster. Its name owes homage to the peregrine falcon, the fastest animal on the planet. Scientists from the university can use the cluster. It is useful for solving computational problems for which a single PC is not powerful enough.

How to use the cluster

There are numerous ways in which the Peregrine cluster can be accessed and used. The following section provides a method to access the cluster whilst keeping file management relatively simple. If you are stuck on any step or unsure how to proceed, please get in touch with the Parallel Computing course helpdesk.

The entire documentation on how to use Peregrine can be found [here](#).

SSH

Peregrine can be reached through a protocol called Secure Shell (SSH). This protocol is sufficient to code on the cluster and test the performance of your different programs. To establish an ssh connection with the Peregrine cluster, a simple command line or VSCode can be used.

- In the case of a command line, simply type `ssh fxxxxxx@pg-interactive.hpc.rug.nl`, with `fxxxxxx` replaced with the username you retrieved. This command will establish an ssh connection with the Interactive Peregrine node, a node specifically designed for running small code snippets with less than a few minutes of runtime - such as compiling your code or doing short performance tests.
- An ssh connection can also be established via VSCode, allowing you to work fully remotely on the server while staying in your familiar editor. To do so, install the 'Remote-SSH' extension in VS Code, then select `Remote-SSH: Connect to Host...` from the Command Palette (`F1` or `Ctrl`+`⇧`+`P`) and use the `pg-interactive.hpc.rug.nl` hostname.

When ssh'ed into the Peregrine cluster, you have the same capabilities as you do on your local machine (either through the terminal or VSCode). With this, you can create your own files, edit them, and compile/execute them.

Working Locally

It is however possible to work locally on your machine and to copy over files into the cluster for execution and testing. There exist many methods for copying files over secure shell, and below two such methods are provided.

- The simpler, first method is called `rsync`. This method allows one to copy local files directly to a host over any remote shell.

- A more complicated, but more convenient method is called `sshfs`, which stands for SSH File System. To mount the Peregrine cluster to your local directory, first create a folder where you want the files to be mounted, and then run the following command:

```
sudo sshfs -o allow_other,default_permissions fxxxxx@pg-interactive.hpc.rug.nl:~/[foo] [bar]
```

, where `[foo]` is the folder on peregrine you want to make available locally, and `[bar]` is the destination for this folder on your local drive.
 - Using this method, you can still choose between editing your files in this folder directly, or periodically copying over your files from another folder to this one. In general, as the files are on the Peregrine system and not your local machine, make sure to create a local copy periodically.
- After copying the files to Peregrine, one can `ssh` to the Peregrine login node in interactive mode (see section on SSH).

Executing Code

In the previous sections, we showed several ways to work with the Peregrine cluster, by either directly editing on it, or transferring your local files. The final detail which needs to be tackled is how to execute code on the cluster. Fortunately, this process is rather simple.

- The first step consists of executing `module load foss/2021a`. This command is required to make sure the latest gcc compiler is used.
- From this point onwards, you should be able to compile and run your code as you would on your local machine.