

# Our Workshop Environment

John Urbanic  
Parallel Computing Scientist  
Pittsburgh Supercomputing Center

# Our Environment Today

Your laptops or workstations: only used for portal access.

Bridges-2 is our HPC platform.

We will here briefly go through the steps to login, edit, compile and run before we get into the real materials.

We want to get all of the distractions and local trivia out of the way here. Everything *after* this talk applies to any HPC environment you will encounter.



# Getting Connected

- The first time you use your account sheet, you must go to [apr.psc.edu](http://apr.psc.edu) to set a password. We will take a minute to do this shortly.
- We will be working on [bridges2.psc.edu](http://bridges2.psc.edu). Use an ssh client (a Putty terminal, for example), to ssh to the machine.
- At this point you are on a login node. It will have a name like “bridges2-login011”. This is a fine place to edit and compile codes. However we must be on compute nodes to do actual computing. We have designed Bridges to be the world’s most interactive supercomputer. We generally only require you to use the batch system when you want to. Otherwise, you get your own personal piece of the machine. To get a single GPU use “interact -gpu”:

```
[urbanic@bridges2-login011]$ interact -gpu  
[urbanic@v005]$
```



- However when we have hundreds of you looking for very quick turnaround, we will fall back on the queuing system to help. We will keep it very simple today:

```
[urbanic@bridges2-login011]$ sbatch gpu.job
```

# Editors

For editors, we have several options:

- emacs
- vi
- nano: use this if you aren't familiar with the others

# Compiling

We will be using standard Fortran and C compilers. They should look familiar.

- `nvc` for C
- `nvfortran` for Fortran

Note that on Bridges you would normally have to enable this compiler with

```
module load nvhpc
```

I have put that in the `.bashrc` file that we will all start with.

# Multiple Sessions

**There is no reason not to open other sessions (windows) to the login nodes for compiling and editing. You may find this convenient. Feel free to do so.**

# Our Setup For This Workshop

After you copy the files from the training directory, you will have:

```
/Exercises
    /Test
    /OpenMP
        laplace_serial.f90/c
    /Solutions
    /Examples
    /Prime
/OpenACC
/MPI
```

# Preliminary Exercise

Let's get the boring stuff out of the way now.

- Log on to `apr.psc.edu` and set an initial password.

- Log on to Bridges.

```
ssh username@bridges2.psc.edu
```

- Run the setup script that will copy over the Exercises directory we will all use. It will also automatically load the right compiler using your `.bashrc` script whenever you login.

```
~training/Setup
```

- As told, logout and log back on again to complete the setup. You won't need to do that in the future.

- Edit a file to make sure you can do so. Use `emacs`, `vi` or `nano` (if the first two don't sound familiar).

- `cd` into your exercises/test directory and compile (C or Fortran)

```
cd Exercises/Test
nvc test.c
nvfortan test.f90
```

- Run your program

```
sbatch gpu.job
```

(Wait a minute, or see how your job is doing with `squeue -u username`)

- Look at the results

```
more slurm-55838.out
```

(The exact job number will differ)

It should say "Congratulations!"