Our Workshop Environment

John Urbanic
Parallel Computing Scientist
Pittsburgh Supercomputing Center
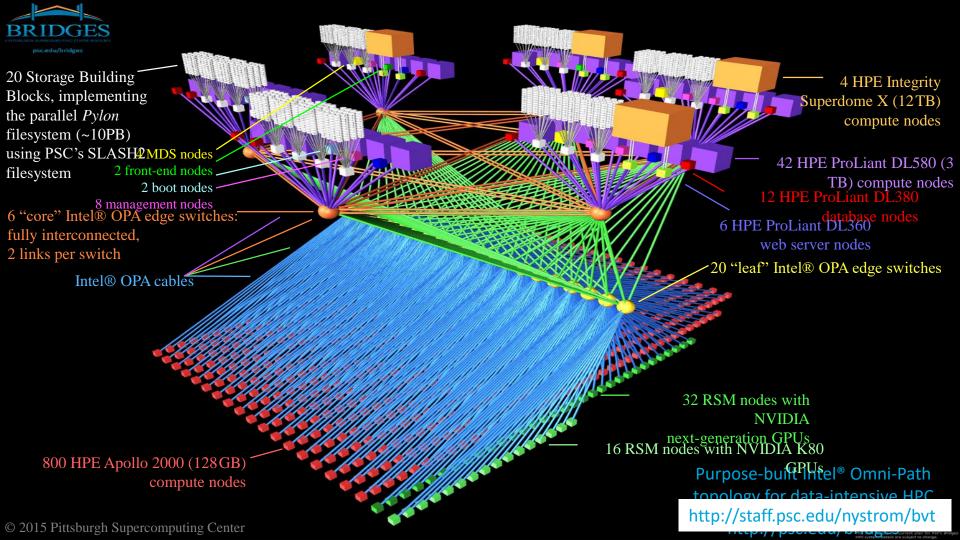
Our Environment This Week

- **Your laptops or workstations: only used for portal access**

- **Bridges 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.

BRIDGES
A PITTSBURGH SUPERCOMPUTING CENTER RESOURCE
psc.edu/bridges

20 Storage Building Blocks, implementing the parallel *Pylon* filesystem (~10PB) using PSC's SLASH2 filesystem

2 MDS nodes
2 front-end nodes
2 boot nodes
8 management nodes

6 "core" Intel® OPA edge switches: fully interconnected, 2 links per switch

Intel® OPA cables

4 HPE Integrity Superdome X (12 TB) compute nodes

42 HPE ProLiant DL580 (3 TB) compute nodes

12 HPE ProLiant DL380 database nodes

6 HPE ProLiant DL360 web server nodes

20 "leaf" Intel® OPA edge switches

32 RSM nodes with NVIDIA next-generation GPUs

16 RSM nodes with NVIDIA K80 GPUs

800 HPE Apollo 2000 (128 GB) compute nodes

Purpose-built Intel® Omni-Path topology for data-intensive HPC

http://staff.psc.edu/nystrom/bvt

http://psc.edu/bridges

Bridges Node Types

Type	RAM	Phase	n	CPU / GPU / other	Server
ESM	12TB[b]	1	2	16 × Intel Xeon E7-8880 v3 (18c, 2.3/3.1 GHz, 45MB LLC)	HPE Integrity Superdome X
ESM	12TB[c]	2	2	16 × Intel Xeon E7-8880 v4 (22c, 2.2/3.3 GHz, 55MB LLC)	HPE Integrity Superdome X
LSM	3TB[b]	1	8	4 × Intel Xeon E7-8860 v3 (16c, 2.2/3.2 GHz, 40 MB LLC)	HPE ProLiant DL580
LSM	3TB[c]	2	34	4 × Intel Xeon E7-8870 v4 (20c, 2.1/3.0 GHz, 50 MB LLC)	HPE ProLiant DL580
RSM	128GB[b]	1	752	2 × Intel Xeon E5-2695 v3 (14c, 2.3/3.3 GHz, 35MB LLC)	HPE Apollo 2000
RSM-GPU	128GB[b]	1	16	2 × Intel Xeon E5-2695 v3 + 2 × NVIDIA Tesla K80	HPE Apollo 2000
RSM-GPU	128GB[c]	2	32	2 × Intel Xeon E5-2683 v4 (16c, 2.1/3.0 GHz, 40MB LLC) + 2 × NVIDIA Tesla P100	HPE Apollo 2000
DB-s	128GB[b]	1	6	2 × Intel Xeon E5-2695 v3 + SSD	HPE ProLiant DL360
DB-h	128GB[b]	1	6	2 × Intel Xeon E5-2695 v3 + HDDs	HPE ProLiant DL380
Web	128GB[b]	1	6	2 × Intel Xeon E5-2695 v3	HPE ProLiant DL360
Other[a]	128GB[b]	1	16	2 × Intel Xeon E5-2695 v3	HPE ProLiant DL360, HPE ProLiant DL380
Gateway	64GB[b]	1	4	2 × Intel Xeon E5-2683 v3 (14c, 2.0/3.0 GHz, 35MB LLC)	HPE ProLiant DL380
Gateway	64GB[c]	2	4	2 × Intel Xeon E5-2683 v3	HPE ProLiant DL380
Storage	128GB[b]	1	5	2 × Intel Xeon E5-2680 v3 (12c, 2.5/3.3 GHz, 30 MB LLC)	Supermicro X10DRi
Storage	256GB[c]	2	15	2 × Intel Xeon E5-2680 v4 (14c, 2.4/3.3 GHz, 35 MB LLC)	Supermicro X10DRi
Total	**281.75TB**		**908**		

a. Other nodes = front end (2) + management/log (8) + boot (4) + MDS (4)
b. DDR4-2133
c. DDR4-2400

Getting Connected

- The first time you use your account sheet, you must go to apr.psc.edu to set a password. You may already have done so, if not, we will take a minute to do this shortly.

- We will be working on bridges.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 "br001" or "br006". 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 8 processors on a node use "interact":

```
[urbanic@br006 ~]$ interact -n 8
[urbanic@r590 ~]$
```

- You can tell you are on a regular memory compute node because it has a name like "r590" or "r101".

- Do make sure you are working on a compute node. Otherwise your results will be confusing.

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 this week. They should look familiar.

- **pgcc (aka mpicc) for C**
- **pgf90 (aka mpif90) for Fortran**

We will slightly prefer the PGI compilers (the Intel or gcc ones would also be fine for most of our work, but not so much for OpenACC). There are also MPI wrappers for these called mpicc and mpif90 that we will use. Note that on Bridges you would normally have to enable this compiler with

```
module load pgi
```

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

Multiple Sessions

You are limited to one interactive compute node session for our workshop. However, 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
        /OpenACC
        /MPI
                laplace_serial.f90/c
                laplace_template.f90/c
                /Solutions
                /Examples
```

Preliminary   Exercise

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

- **Log on to apr.psc.edu and set an initial password if you have not.**

- **Log on to Bridges.**
  ssh username@bridges.psc.edu

- **Copy the exercise directory from the training directory to your home directory, and then copy the workshop shell script into your home directory.**
  cp -r ~training/Exercises .
  cp ~training/.bashrc .

- **Logout and back on again to activate this script. 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).**

- **Start an interactive session.**
  interact -n 8

- **cd into your exercises/test directory and compile (C or Fortran)**
  cd Exercises/Test
  pgcc test.c
  pgf90 test.f90

- **Run your program**
  a.out   **(You should get back a message of "Congratulations!")**