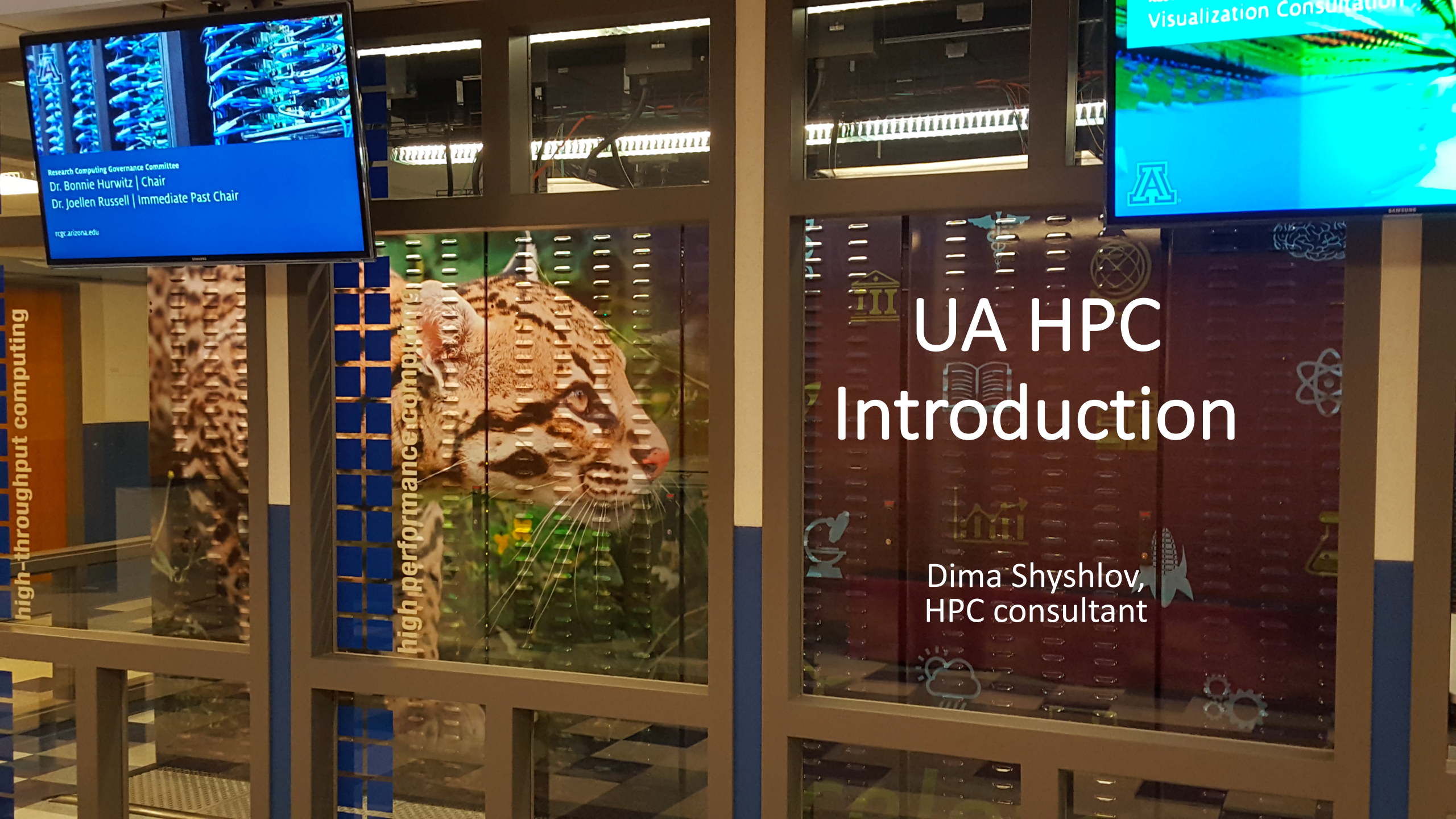



# UA HPC Introduction

Dima Shyshlov,
HPC consultant

# Requesting an HPC account

- accounts.arizona.edu

  -> manage your accounts

  -> HPC account

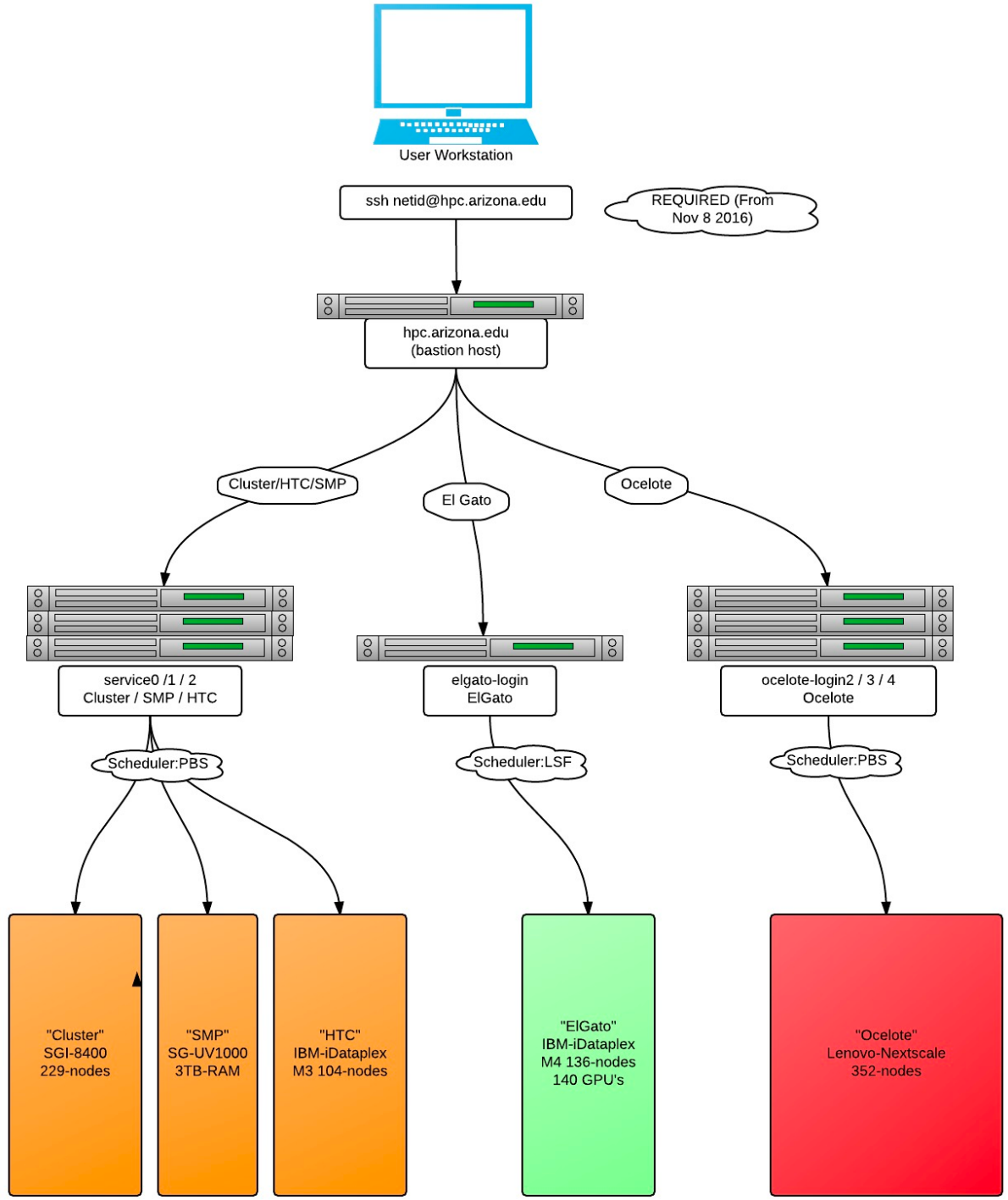
  -> Notify your sponsor of your request at the HPC sponsorship page

# Bastion host

```
dhcp-10-132-143-170:~ dshyshlov$ ssh dshyshlov@hpc.arizona.edu
Password:
Duo two-factor login for dshyshlov


Enter a passcode or select one of the following options:


 1. Duo Push to XXX-XXX-0896
 2. Phone call to XXX-XXX-0896
 3. SMS passcodes to XXX-XXX-0896 (next code starts with: 2)


Passcode or option (1-3): 1
Success. Logging you in...
Last login: Mon Aug 28 14:20:47 2017 from dhcp-10-132-143-170.uawifi.arizona.edu
This is a bastion host used to access the rest of the environment.


Shortcut commands to access each resource
-----------------------------------------
Ocelote:                El Gato:                Cluster(ICE)/HTC/SMP:
$ ocelote               $ elgato                $ ice
```

# HPC systems in UofA

- Legacy systems (Ice):
  - Cluster
  - HTC
  - SMP
- El Gato
- Ocelote:
  - over 300 compute nodes
  - 28 cores per node
  - 6GB of memory per core



# File transfer

- There are special nodes for data transfer
  - sftp.hpc.arizona.edu
- Connecting to file transfer node
  - sftp NetID@sftp.hpc.arizona.edu
- File transfer software
  - WinSCP (Windows), Cyberduck (Windows and Mac), Fugu (Mac)
- Other ways of file transfer:
  - Globus (large files), scp, rsync, irods

# Login nodes VS Compute nodes

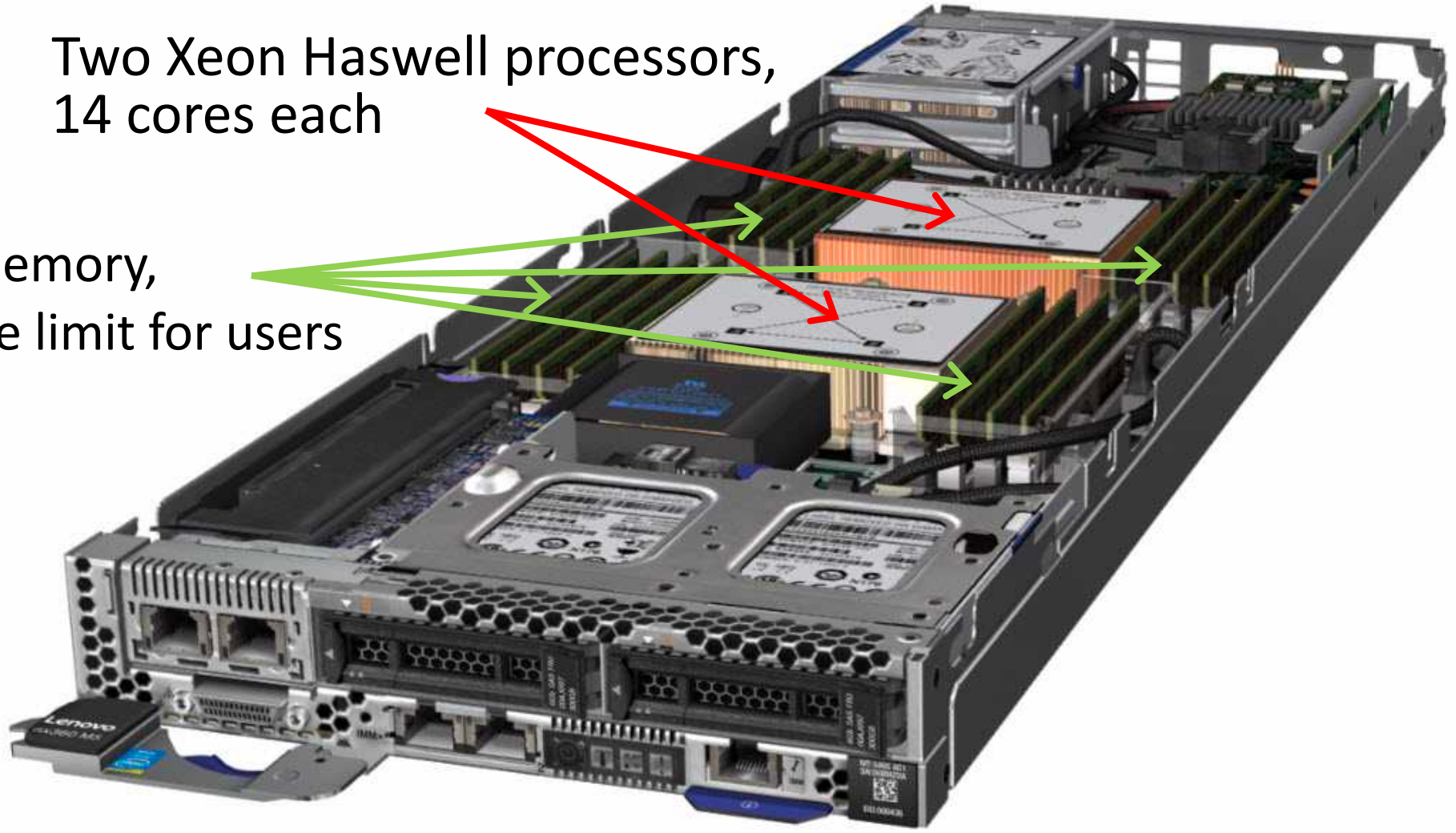
- Login nodes are for:
  - editing code, scripts
  - submitting jobs (calculations)
  - checking status of the jobs
  - testing and troubleshooting
  - interactive tasks
  - Ocelote has 3 login nodes

- Compute nodes are for running jobs
  - ideal for batch jobs
  - Ocelote has 331 compute nodes

# Anatomy of a computing node

Two Xeon Haswell processors,
14 cores each

192 GB of memory,
168 GB is the limit for users



# Storage and Allocation

- Storage:
  - Home directory – 15GB
  - /extra – 200GB
  - /xdisk – temporary storage up to 1TB
  - /rsgrps – rented storage by research groups
  - *uquota* – Linux command to display your used/available storage

- Allocation
  - standard – limited to 24,000 hours/group/month
  - windfall – unlimited, jobs can be preempted
  - *va* – Linux command to display available allocation

# Software

- Many software packages are available as modules
  - *module avail* – list all the installed modules
  - *module avail python*– list all versions of python
  - *module load python*– load the module (the latest version is usually the default)
  - *module list* – display all the modules loaded in your environment

# ”Hello, World!” exercise

- Copy exercise files:
  - *git clone https://github.com/dshyshlov/UA-HPC-Intro.git*
- List the files and directories:
  - *ls*
- Change directory to UA-HPC-Intro
  - *cd UA-HPC-Intro* (use tab for autocompletion)
- List the files again:
  - *ls*

# mpi_hello_world.c

- Multicore version of “Hello, World!” program in C language
  - Uses MPI to run on multiple nodes

- Enable using MPI with:
  - *module load openmpi*

- Compile with:
  - *mpicc -o mpi_hello_world mpi_hello_world.c*

# PBS Script

- Parameters for scheduler
- Loading necessary software
- Navigating to the working directory
- Run the program

```
#!/bin/bash
#PBS -N mpi_hello_world
#PBS -W group_list=hpcteam
#PBS -q windfall
#PBS -l select=1:ncpus=1:mem=6gb:pcmem=6gb
#PBS -l walltime=0:1:0
#PBS -l cput=0:1:0


module load openmpi


cd ~/UA-HPC-Intro/


mpirun -n 1 ./mpi_hello_world
```

# PBS Script

- Display the content of the PBS script on the screen:
  - *cat script.pbs*
- Edit the PBS script with nano text editor:
  - *nano script.pbs*
- Fill in the group name
- Submit the script with the command:
  - *qsub script.pbs*
- Check the job status:
  - *qstat –u NetID*

# Output and Error files

- Check the output file
- Check the error file
- Output and error files can be joined together with the PBS script:
  - #PBS –j oe
- You can also specify the file names:
  - #PBS –o output.txt
  - #PBS –e error.txt

# ”Hello, World!” on 10 cores

- Make a second copy of the PBS script:
  - cp script.pbs 10cores_script.pbs
- Edit 10cores_script.pbs to run the “Hello, World!” code on 10 cores
- What parameters you need to change?
- Submit the job with 10cores_script.pbs

# Getting help

- docs.hpc.arizona.edu
- hpc-consult@list.arizona.edu
- Google, stackoverflow…