

# Computational Cluster

## Basic Usage with Slurm

Alex Heilman

August 12, 2025

# Overview

Computational  
Cluster

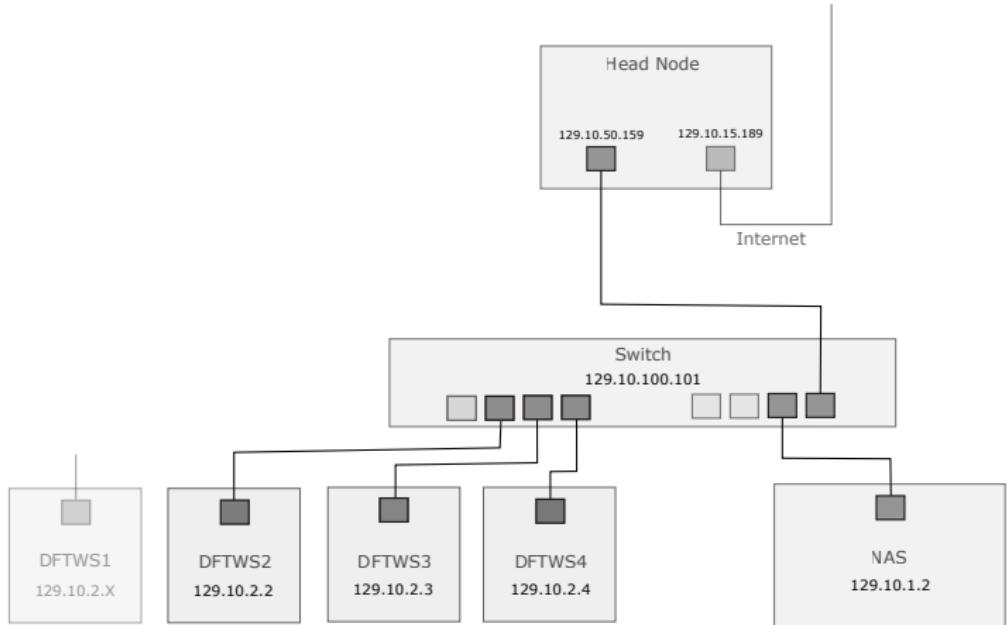
Alex Heilman

- Network architecture & access
- Running commands across multiple machines with Ansible
- Slurm usage & troubleshooting

# Architecture

## Computational Cluster

Alex Heilman



All devices are on subnet of the gateway HeadNode

# What's a subnet?

- A subnet is assigned a subnet mask
- All IPs under it's mask are on the subnet
- Subnet masks are specified by the gateway's IP address,  
ours is 255.255.0.0 or 129.10.0.0/16
- This means all nodes must have IP in range 129.10.x.x

# What's a gateway?

- A gateway gives a LAN access to the larger internet.
- The head node is configured with `iptables`
- For new nodes, pass this command locally to set the default gateway:

```
sudo route add default gw 129.10.50.159
```

# How to Access the Cluster

The worker nodes are on the subnet 129.10.X.X with the gateway set as 129.10.15.189 (the head node).

- To access these nodes, use the -J option with SSH to jump through the head node.

```
ssh -J <user>@129.10.15.189 <user>@<subnode-IP>
```

- Append rule to “`~/.ssh/config`” file to automatically jump with ProxyJump

# What's a DNS server?

- A Domain Name Service (DNS) provides maps from IP to web addresses.
- DNS server is often set dynamically by network admins, we need to specify ourselves for subnodes.
- We can use Google's public DNS 8.8.8.8
- If broken, for quick fix:

```
echo "nameserver 8.8.8.8" | sudo tee /etc/resolv.conf > /dev/null
```

# Mounting the NAS

- We mount with NFS
- Must install nfs-common on all devices
- Add line to /etc/fstab:

```
<NAS-IP>/<NAS-Shared-Folder>  <Directory-to-Mount-to>    nfs    defaults    0 0 :
```

- Confirm /etc/fstab isn't broken with  
`findmnt --verify`
- Reload /etc/fstab with `mount -a`

# What's Ansible?

Ansible allows us to run commands across several machines



- State vs. action
- Idempotency

# Defining Inventory/Ansible Hosts

- Must define host and group names of devices to run ansible commands
- These are specified in local inventory files or globally via /etc/ansible/hosts

```
[HEAD]
129.10.15.189

[DFTWS]
129.10.2.2
129.10.2.3
129.10.2.4

[DATABASE]
129.10.1.2
```

The current hosts file is shown above.

# Host Prerequisites

- Before running ansible, all hosts must be accessible by SSH without prompts.
- Generate RSA key pairs with command on head node:

```
ssh-keygen -t rsa
```

Then copy SSH keys to subnodes with:

```
ssh-copy-id <user>@<subnode-identifier>
```

After this, no password should be required.

## Ex: Installing packages cluster-wide

- We can use ansible to install packages across all devices:

```
---
```

```
- hosts: all
```

```
tasks:
```

```
- name: Install packages
```

```
apt:
```

```
state: present
```

```
name:
```

```
- ntp
```

```
- htop
```

```
- vim
```

We can run the above 'site.yml' playbook with:

```
ansible-playbook --ask-become-pass site.yml
```

where the flag --ask-become-pass will prompt for sudo password of current user.

# Ex: Setting up environments cluster-wide

- We can also use ansible to set up pip environments:

```
- name: install python packages with pip
become: yes
pip:
name:
- wheel
- flask
- gunicorn
virtualenv: '/myproject/myprojectenv'
```

# What's a daemon?

Daemons run in the background and process requests to services that run continuously, waiting for input

```
ajh@HeadNode:/mnt/nas$ systemctl status slurmctld
● slurmctld.service - Slurm controller daemon
  Loaded: loaded (/etc/systemd/system/slurmctld.service; enabled;>)
  Active: active (running) since Sun 2025-08-03 23:19:12 EDT; 1 w>
    Main PID: 846046 (slurmctld)
      Tasks: 78 (limit: 76296)
        Memory: 11.8M
          CPU: 10min 17.753s
        CGroup: /system.slice/slurmctld.service
                  ├─846046 /opt/slurm-25.05.0-build2/sbin/slurmctld --sys>
                  └─846080 "slurmctld: slurmscriptd" "" "" "" "" "" "" "" "
```

- System daemon controls other daemons through it's own: systemd.
- Start, stop, and enable services via systemctl
- On reboot, slurm daemons must be reset manually with

```
systemctl start <daemon-name>
scontrol update NodeName=<node-name> state=resume
```

# What's Slurm?

Computational  
Cluster

Alex Heilman

Slurm allows us to schedule jobs across the cluster



- Head node runs `slurmctld` daemon
- Worker nodes runs `slurmd` daemon

# Using Slurm

Computational  
Cluster

Alex Heilman

- Logs are located at `\var\log\<daemon-name>.log`
- Configuration file at `\etc\slurm\slurm.conf`
- Try locating scripts and data needed to run in `/mnt/nas`

# Using Slurm

- Check cluster status with sinfo

```
(base) ajh@HeadNode:~$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
slurm*      up    infinite     1  idle* DFTWS3
slurm*      up    infinite     2  idle DFTWS[2,4]
```

- Check specific node with scontrol show node <name>

```
(base) ajh@HeadNode:~$ scontrol show node DFTWS2
NodeName=DFTWS2 Arch=x86_64 CoresPerSocket=1
CPUAlloc=0 CPUEfctv=1 CPUTot=1 CPULoad=1.90
AvailableFeatures=(null)
ActiveFeatures=(null)
Gres=(null)
NodeAddr=DFTWS2 NodeHostName=DFTWS2 Version=25.05.0
OS=Linux 6.8.0-65-generic #68~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC
MIC Tue Jul 15 18:06:34 UTC 2
RealMemory=1 AllocMem=0 FreeMem=160686 Sockets=1 Boards=1
State=IDLE ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_
Label=N/A
Partitions=slurm
BootTime=2025-08-07T20:05:56 SlurmdStartTime=2025-08-10T21:54
:48
LastBusyTime=2025-08-11T19:22:28 ResumeAfterTime=None
CfgTRES=cpu=1,mem=1M,billing=1
AllocTRES=
CurrentWatts=0 AveWatts=0
```

# Using Slurm

Computational  
Cluster

Alex Heilman

- Run command on cluster with `srun <command>`
- Submit script to scheduler with `sbatch <script>`
- Check queue of jobs with `squeue`
- Cancel job or allocation with `scancel <job-id>`

# Recap

- Network architecture & access
  - ▶ Need to jump through head node (gateway)!
  - ▶ DNS server may be reset occasionally by restart
  - ▶ NAS is mounted globally to `\mnt\nas`
- Can use Ansible for cluster-wide commands
  - ▶ Install packages
  - ▶ Create environments
  - ▶ Setup Slurm
- Schedule large jobs with Slurm
  - ▶ Try to run from `\mnt\nas`
  - ▶ Check status of cluster with `sinfo`
  - ▶ Check status of jobs with `squeue`
  - ▶ Use `sbatch` for scripts and `srun` for commands

# Recap

- Problems?
  - ▶ Confirm \$PATH contains Slurm bin directory
  - ▶ Check node with scontrol show node
  - ▶ Check daemons with systemctl status
  - ▶ Check logs at \var\log
  - ▶ Maybe contact me