

August 2023

Connect to ARC systems and run your first jobs

Matthew Brown
Computational Scientist
Advanced Research Computing



ARC Course Offerings

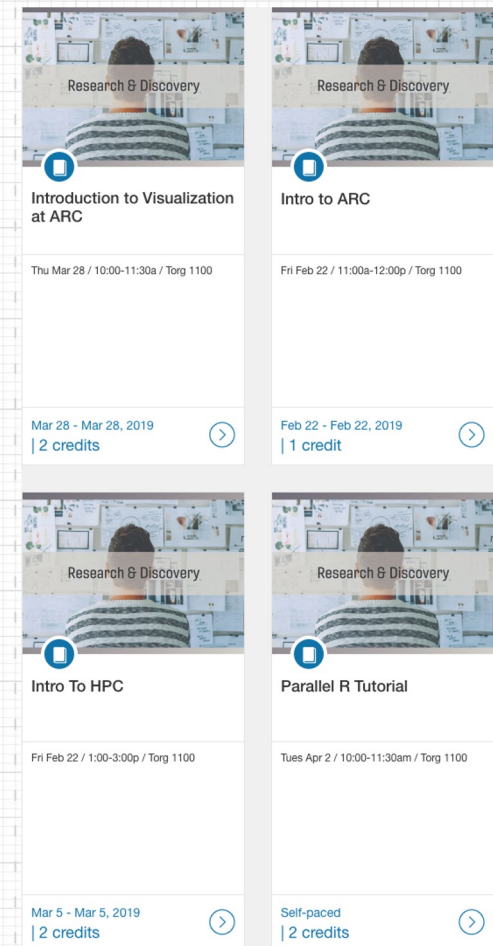
Connect to ARC systems and run your first jobs

- Thursday, August 17th, 2023 1:30 - 2:45 (Zoom)
- Wednesday, October 4th, 2023 3:30 - 4:45 (Zoom)

This workshop is geared towards VT faculty, researchers, and students who are new to ARC. The aim is to provide orientation to the user-facing components of ARC systems and to demonstrate common connection and usage patterns.

This includes an overview and demonstration of ARC's web-based portal (Open OnDemand), and also how to connect with command-line oriented tools. Attendees with ARC accounts can follow along in a walkthrough of the most useful scheduler-interaction commands and an overview of building and submitting a sample workload, to the scheduler in the form of a batch job.

- Connect via Open OnDemand
- Connect via SSH
- Cluster and Scheduler Orientation
- Run Demo Jobs



Fall 2023 ARC workshop Series

Introductory / Orientation Workshops (75 minutes each):

- *Advanced Research Computing (ARC) Overview*
- *Connect to ARC systems and run your first jobs*
- *Get your software/code to run on ARC*

Series 1: 8/17 - 8/18

Series 2: Wednesday afternoons: 9/27, 10/4, 10/11

Special Topics (75 minutes each):

- *Monitoring Resource Utilization and Job Efficiency*
- *Getting the Best Data Storage Performance on ARC Filesystems*
- *Launching in Parallel*

Series 1: Biweekly, Tuesday mornings 9/5, 9/19, 10/3

Series 2: Weekly, Thursday afternoons 10/26, 11/2, 11/9



Expectations

- This is an informal workshop
- Mostly informational about ARC and research computing at VT
- I want to hear your questions
- Welcome to use chat to ask questions + some time at the end
- Feedback needed to help improve future workshops
 - One up / one down at the end

First Thing's First

VPN needed for connections from off-campus

- <https://onecampus.vt.edu/task/all/installing-pulse-vpn>
- Nearly all ARC services require being on the campus network or VPN
- Use “VT Traffic over SSL VPN” connection vs. “All Traffic over SSL VPN”
- ColdFront (accounting system) available with or without VPN

Get an ARC account:

- <https://coldfront.arc.vt.edu/account/create>
- Acceptable Use Policy



Getting Started

https://www.docs.arc.vt.edu/get_started.html

Needs Assessment

- Compute
- Storage
- Software
- Collaboration
- Visualization
- Lifecycle and data retention

Get an account

<https://arc.vt.edu/account>

- Get account for log-in

Register a Project and Get Allocations

<https://coldfront.arc.vt.edu>

- Create a “project”, add people, grants/pubs
- Request allocation for Compute to run jobs
- Request allocation for Project storage if desired

Where to get help

Website (<https://docs.arc.vt.edu>)

- FAQs
- Video demos
- Detailed instructions
- Examples

<https://github.com/AdvancedResearchComputing/example>

Helpdesk (<https://arc.vt.edu/help>)

Office Hours (<https://arc.vt.edu/office-hours>)

Ask for consultation

- Workflow design
- Optimization
- Sponsored Projects



High Performance Computing

ARC hosts several systems designed for high-performance and/or high-throughput computing (HPC/HTC)

| | | |
|-------------------|--|-------------------------------|
| CUI | Dense GPU + some CPU for projects with controlled data/software | c. 2021 |
| Tinkercliffs | HPC/HTC, Flagship CPU, Cost Center Capable AI/ML Dense GPU nodes more Dense GPU nodes | c. 2020 c. 2021 c. 2022 |
| Infer | Accelerating inference and AI workloads | c. 2020 |
| OWL (coming soon) | Water-cooled, latest generation AMD CPU, high mem-per-core, DDR5 | c. 2023 |
| Coming next | GPU node expansion | c. 2024 |

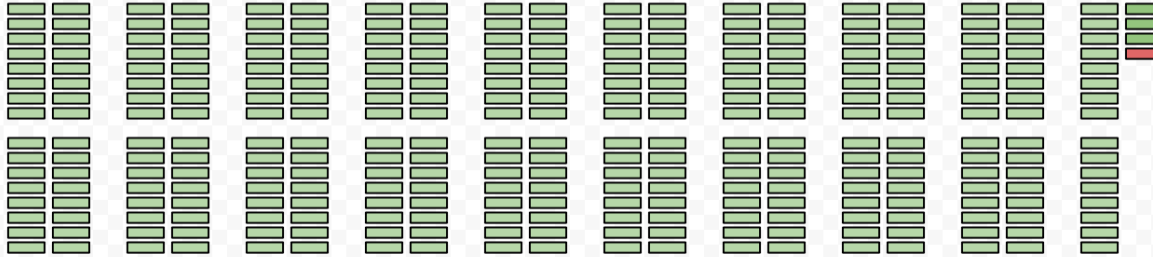
TinkerCliffs - Flagship CPU Cluster

316 Nodes w/ 128 cores(AMD EPYC Rome)
 16 Nodes w/ 96 cores (Intel Cascade Lake-AP)
 41,984 CPU cores

tc-hm[001-008]
 largemem_q



tc[001-308]
 dev_q, preemptable_q
 tc[001-307]
 normal_q
 tc[001-302]
 interactive_q
 tc308

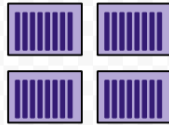


tc-intel[001-016]



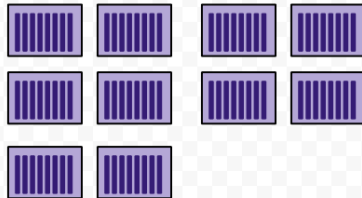
w/ dense GPU

ai[001-04]
 a100_normal_q



4 Nodes w/ 128 cores (AMD Epyc Rome 7742)
 + 8 NVIDIA A100-80GB GPUs (6912 CUDA)
 512 CPU cores
 32 GPU accelerators
 221,184 CUDA cores

ai[001-04]
 a100_normal_q



10 Nodes w/ 128 cores (AMD Epyc Rome 7742)
 + 8 NVIDIA A100-80GB GPUs (6912 CUDA)
 1280 CPU cores
 80 GPU accelerators
 552,960 CUDA cores

Soon: 2022 expansion



Web Access to Clusters

Open OnDemand

- Connect to VPN first
- Manage data: graphical interface to browse, transfer, view, and edit files
- Access the shell command-line interface (CLI) on ARC
- Interactive jobs, especially applications with graphical user interfaces (Matlab, Rstudio, Jupyter Notebooks, ...)

<https://ood.arc.vt.edu>

Overview of SSH/CLI Connection

-- Must first be on campus network or VPN

Linux or MacOS

- Use built-in "terminal" application to get CLI with standard tools
- ssh, ssh-keygen, ssh-agent, scp, rsync

Windows: may need to install an SSH client

- Recent versions have SSH CLI readily available
- GUI client downloads: MobaXterm, PuTTY (no longer recommended)
- Git/BASH (Git for Windows with a shell emulator)
- Windows Subsystem for Linux (WSL), Microsoft VS-Code

Windows: pick and install an SSH client

Windows built-in SSH client

- <https://learn.microsoft.com/en-us/windows/terminal/tutorials/ssh>
- “The latest builds of Windows 10 and Windows 11 include a built-in SSH server and client that are based on OpenSSH”
- “You can ... check that it is installed in Windows Settings > Apps > Optional features, then search for "OpenSSH" in your installed features.”

Windows: pick and install an SSH client

-- Free, open-source software. Includes Git for source control --

Git/BASH from Git for Windows

- Lightweight unix shell and tools
- <https://gitforwindows.org>
- Lots of questions to answer during install, but most relate only to the “Git” configuration.
- Default working directory is your Windows user directory: /c/Users/Myaccount



Git is the current standard tooling for version control and collaborative development, so this package has added value.

Also used by Software Carpentries

<https://github.com/git-for-windows/git/releases/download/v2.35.3.windows.1/Git-2.35.3-64-bit.exe>
<https://software-carpentry.org/lessons/>

Windows: GUI SSH clients

-- A traditional way to SSH from Windows, but no longer recommended!

MobaXterm

- ~~MobaXterm: <https://mobaxterm.mobatek.net/download.html>~~
- ~~Integrated X server and GUI file transfer utility~~

~~PuTTY: <https://www.putty.org>~~

- ~~It works and is lightweight~~
- ~~Not as fully integrated with files browser and transfer tool~~
- ~~No X server, strange copy/paste~~

Windows: pick and install an SSH client

Windows Subsystem for Linux

- Essentially a Linux virtual machine running inside Windows
- Bigger download (~250MB)
- Highest level of features, options, etc. A more native Linux experience.

Installation:

- Settings > Apps > Programs and Features > Turn Windows features on or off > Windows Subsystem for Linux OR (latest versions) "wsl --install"
- Reboot
- Install Linux distribution using Microsoft Store (Ubuntu)
- Your Windows files are available in the VM at /mnt/c

<https://docs.microsoft.com/en-us/windows/wsl/install>



Connect via SSH

-- must be on campus network or connect to VPN

```
“ssh myusername@tinkercliffs2.arc.vt.edu”
```

- On initial connection, type “yes” to accept the authenticity of the host and continue connecting
- Password, then DUO for 2-factor authentication

SSH Keys

Alternative mode of two-factor authentication. Provides faster/simpler connections after setup.

- Generate keypair
 - Unix shells: `ssh-keygen`
 - Putty/MobaXterm: key generator GUI (PuTTYgen)
 - Protect your private key!
 - 4096 bit encryption (-b 4096), usually RSA
 - secure with secret passphrase to satisfy VT requirement for 2-factor authentication
- Add contents of public key to a line on this file on ARC
 - `/home/myusername/.ssh/authorized_keys`
 - Use editor in OnDemand to simplify this
 - Also: `ssh-copy-id myusername@tinkercliffs1.arc.vt.edu`

SSH Keys – cont.

-- Alternative mode of two-factor authentication. Provides faster/simpler connections once set up.

SSH-Agent (Linux/unix shells or emulators only)

- Caches any added ssh private keys while your shell is open. Reduces passphrase re-entry.
- Command: "ssh-agent", then, "ssh-add"
- If using Git/BASH, copy-paste output from "ssh-agent" command back into the shell
- Should close shell each time you leave the workstation for best security.

This setup provides quickest authentication process and satisfies two-factor requirement:

1. Private key passphrase (cached by ssh-agent)
2. Public/private key challenge

https://learn.microsoft.com/en-us/windows-server/administration/openssh/openssh_keymanagement



Detach and reattach to sessions

Commands running in remote shells are children of the SSH connection. If the connection is terminated, then so are all child processes. This can be problematic for long-running processes.

Solution:

Linux utility: screen (tmux has similar functionality)

- Run “screen” – opens a new, detachable shell, not a child of the calling shell
- Start commands in shell
- Ctrl+a, d to “detach”
- "screen -r" to resume

SLURM Intro

- SLURM = scheduler and cluster resource manager
- Request resources
 - `salloc` – get an allocation and wait for further commands interactively
 - `sbatch` – submit job script for non-interactive execution
 - `srun` – run command in allocation if the context already exists OR request an allocation and run the command
- Queued until resources are available (enhanced FIFO)
- Partial nodes can be allocated, ie. nodes can be shared by multiple jobs. Each job is contained to its allocation with linux cgroups

Commands: "--help", "man sbatch", find str with "/str". "n", "p", "q"
"srun hostname", squeue, sprio



Anatomy of a Batch Script

```
#!/bin/bash
#SBATCH -J hello-world
#SBATCH --account=arcadm
#SBATCH --partition=normal_q
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=1 # this requests 1 node, 1 core.
#SBATCH --time=0-00:10:00 # 10 minutes
##SBATCH --gres=gpu:1
#SBATCH --account=test
echo "hello world from..."; hostname
echo "Working directory is: "; pwd
echo "Changing directory to $SLURM_SUBMIT_DIR"
cd $SLURM_SUBMIT_DIR
module reset
```



Commands: "sbatch <scriptname>", "man sbatch", scancel <jobid>

Accounting

- Old paradigm (Cascades, Dragonstooth, Newriver, Infer)
- New paradigm (Tinkercliffs and future systems)
 - monthly limit per PI on usage (600,000 units/month)
 - PI may own multiple accounts, but limit will be imposed on aggregate usage
 - enables cost center meeting federal requirements
- Billing weights reflect node/component costs
- Also have policy limits per [user,job,account]

Commands: quota, scontrol show part, squeue, showusage, tcgetusage <accountname>, showqos

<https://coldfront.arc.vt.edu>, <https://www.arc.vt.edu>



Support, Consultation and Collaboration

ARC Helpdesk:

<https://arc.vt.edu/help>

<https://4help.vt.edu> (mention ARC, provide details)

ARC Helpdesk GRAs work as a team to handle most incoming questions/problems.

"How do I setup SSH keys for authentication?" "What can I do to get my job to launch faster?" "Why did my job stop?"

"Is MATLAB available on Huckleberry?" "How can I share my files with my collaborator?"

Escalate to ARC Computational Scientists as needed.

Office Hours (<https://arc.vt.edu/office-hours>)



Thanks for watching and listening!

ARC Website: www.arc.vt.edu

My contact info: Matthew Brown
brownm12@vt.edu

