03 Core Workflow Elements

YouTube: https://youtu.be/wbSHOmGn94Q

Hi, and welcome back to ARCC's training video series.

In this video we will look at some of the core workflow elements you'll need to understand to be able to successfully use our clusters.

This video does assume that you have already watched the "Intro to ARCC Services" and "Intro to HPC Clusters" videos.

So, what is a workflow?

In its simplest form a workflow, which can also be referred to as a pipeline, is a sequence of steps which you will work through from the start to the end of your project, and typically it is a pattern you will repeat over and over.

How you perform and implement your workflow is up to you and will be influenced by your technical skills and the workflow complexity, for instance you might have workflows within workflows. You can perform everything manually from the command line, or maybe automate using a script written in say Bash, Python and/or R (just to name a few languages). Some languages provide specific packages to use such as batchtools for R, or you might use an application like Nextflow. The choice is yours.

The focus of this video is to introduce the core elements you will need to consider, understand and perform, somewhere within your workflow.

We will be introducing names of tools, applications and terminology that you may not have heard of, but the idea is for you to start becoming familiar with them. We will go deeper into them with further videos, or you can begin looking up and exploring yourself.

The core elements we will introduce are:

1. Getting your data on/off the cluster.
2. Setting up your environment to be able to perform your research.
3. Running and submitting your jobs on the cluster.

In most cases you won't be able to do your research without data, so one of the first core elements is how to get your data onto the cluster. Closely tied to this is where you store it.

Are you storing/transferring your data to Alcova? If yes, then you can connect from a Linux/Mac platform using SMB, or on Windows via file explorer, or use Globus within a browser.

If you are instead using your /home, /project and/or /gscratch folders on the clusters, then maybe a tool such as **scp**, **sftp** or **rsync** is good enough for your needs.

There are also options to use from the command line **wget**, or **git pull** to retrieve your data.

If you do have your data on Alcova, then one workflow step you will need to include will involve accessing and moving this data into and out of your project/home folders before using it.

Our clusters are used by a diverse group of researchers from all departments across UW. There are a lot of different software needs, 100s of applications, programming with different languages and compilers, and even where there might be a group of users all using R or Python, they're using different versions and different groups of packages and libraries.

To accommodate this we use a module system (LMOD) that enables a researcher to uniquely set up their environment.

This module system enables a researcher to search what applications, what programming languages and scientific libraries, and what compilers are available (installed) on the cluster. You can see what versions are available, as well as what dependencies something might require. For example, we have versions of R that have been built with the gcc compiler as well as the Intel compiler, you would need to indicate which flavor or R and its dependent compiler you want to use.

Once the researcher has discovered what is available, they then need to load these modules to make them available to use. Until you have loaded what you need, they won't be available.

If something isn't available, there are a number of ways to get it onto the clusters from trying to install it yourself (within appropriate boundaries - for example you will not be provided with sudo permissions), or request ARCC to install it for you.
Once you know how to set up your environment and are ready to run your research, then you're set to actually start using the compute nodes.

To facilitate this we use an application called Slurm which is a Workload Manager.

It provides three key functions:

1. First, it allocates access to appropriate computer nodes specific to your requests for some duration of time. You are able to define the number of compute nodes required, as well as for each node the number of cores, the amount of memory, and if a GPU is required. Every submitted 'job' is associated with a researcher's project, and must define a length of time, which could be a couple of hours, a couple of days, up to a maximum of seven days.
2. It provides a framework for starting, executing, monitoring, and even canceling your jobs.
3. Finally, since the clusters are heavily used you will find times that the resources you need are not currently available. When you submit a job if Slurm can run it immediately, it will. Otherwise it will add it to a queue, and start your job when the necessary resources become available, and then notify you once the job has completed. This it will do automatically, allowing you to say submit a job on a Friday afternoon, come back into the office on Monday to find the 'job complete message' in your inbox.

Slurm also allows you to request **interactive sessions** where you essentially have a "live session" directly on a compute node (if one is available). You typically use this to test and debug your work with a subset of data for a short period of time, say a couple of hours. Once you're happy everything appears to be working correctly, you'll then increase the amount of data, size of the simulation, and submit your job to the cluster.

Please note, that although you use the login nodes as an entry point onto the clusters, you should not run any form of computation on them. You must either start an interactive session, or submit a job. By running things on the login nodes you can affect the work of everyone else currently using or trying to access that particular cluster.

To summarize, in this video we have introduced three core elements that you will need to consider within your workflow:

1. Getting your data on/off the cluster.
2. Setting up your environment to be able to perform your research.
3. Submitting your job to the cluster.

There will be names and terms you might not have come across, so head over to our wiki and try searching for them.