Workflow Management on SubMIT

Chad Freer,

Christoph Paus,

Zhangqier Wang,

Luca Lavezzo

Massachusetts Institute of Technology

Jan 2022

SubMIT Machines

SubMIT is a single login computing environment:

- Allows Physics researchers to access heterogeneous high-performance resources
- Access to High Performance Computing (HPD) from department and affiliated labs
- Anybody from the physics department (and friends) invited to work on SubMIT
- Please check out the SubMIT User's Guide for more information
  - **User's Guide**: Link

Some info about SubMIT:

- O(10) powerful server machines: submitXX.mit.edu
- 100 Gb Network switch
- User storage provided O(100 GB)
- Dedicated ultrafast NVMe storage server (O 10 TB) to be used as cache
- Access to hadoop storage system O(10 PB)
- Multiple GPU servers ordered and being prepared for commissioning soon
- Access to multiple computing clusters for batch submission

Running on SubMIT

Once users can log on to the SubMIT machines and start working directly:

- Set up ssh key through submit-portal **Link**

- Load balanced log in:
  ssh <username>@submit.mit.edu

- Account created automatically on SubMIT machines with space available
  1. 5 GB: /home/submit/<username>
  2. 50 GB: /work/submit/<username>

- Can start working directly in these areas
  1. Code stored in your home directories
  2. Larger files to be stored in work area

- Access to public and public_html areas to share
  1. /home/submit/<username>/public
  2. /home/submit/<username>/public_html **Link**

- SubMIT inherently has a lot of functionality (gcc, python, java, etc) **Link**
  1. SubMIT is a shared tool so use responsibly **Link**
  2. You are in charge of maintaining your user areas

Environments

SubMIT has support for many different Environments/Workflows:

- SubMIT inherently has a lot of functionality (gcc, python, java, etc) **Link**
- Some Users will need more complex setups
  1. Specific languages (ROOT, Julia, pythia, etc)
  2. Packages that aren't inherent on SubMIT (coffea, RDataFrame, sklearn, etc)
- Several ways to customize your setup **Link**
  1. CernVM File System (CVMFS) **Link**
     - CVMFS provides a reliable software distribution service
     - Available on Submit at /cvmfs
     - CMSSW: *source /cvmfs/cms.cern.ch/cmsset_default.sh*
  2. Conda **Link**
     - Conda is an open source package management system
     - Gives users full control of their environment
     - Work with MIT JupyterHub Link
     - Once set up: *conda active My_Env*
  3. Containers **Link**
     - Docker Containers deliver software packages called containers
     - Many available in the form of singularity images through CVMFS
     - Coffea:

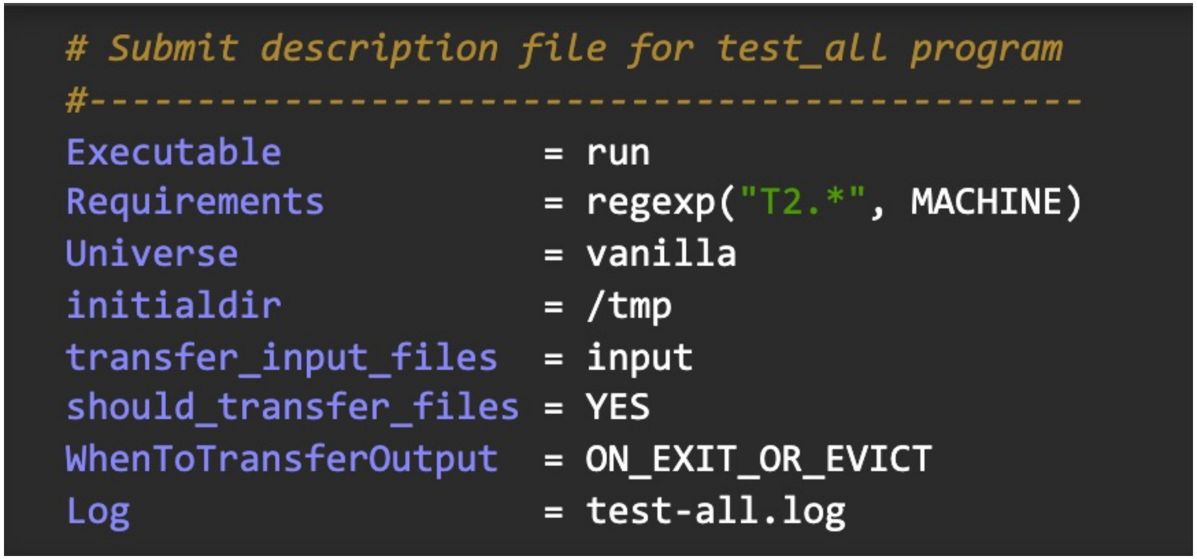
singularity /cvmfs/unpacked.cern.ch/registry.hub.docker.com/coffeateam/coffea-dask:latest

- User's should find the workflow that works best for their applications

HTCondor

Users can scale their projects to analyze thousands of files through HTCondor:

- HTCondor is a batch submission program **Link**
- Once a user application is created and tested on SubMIT machines we can scale
- Will need a condor.sub file which will execute a set of commands you specify
- Examples are shown here: **Link**



```
# Submit description file for test_all program
#-----------------------------------------------
Executable               = run
Requirements             = regexp("T2.*", MACHINE)
Universe                 = vanilla
initialdir               = /tmp
transfer_input_files     = input
should_transfer_files    = YES
WhenToTransferOutput     = ON_EXIT_OR_EVICT
Log                      = test-all.log
```

- This will create a condor job which will execute a script named “run”
- You can customize your job to your liking
  1. Environment
  2. Input files (which files does your executable need in order to run)
  3. Output
  4. Which resources to use