University of Namibia High Performance Computer How-to Guide

Jimmy Shapopi

March 15, 2022

Contents

1 Submitting computing jobs to cluster

The cluster uses the Portable Batch System (PBS) to handle job submissions. This section will detail the basics of job submissions.

There are a few PBS commands that are commonly used and are advantageous to remember:

1. qsub: Submit a job
2. qdel: Delete a batch job
3. qrerun: Rerun a batch job
4. qstat: Show status of batch jobs
5. pbsnodes: List the status and attributes of all nodes in the cluster.

Job submissions are handled by a bash script detailing the commands normally used when operating from a bash terminal. This bash script would also have details such as:

1. Job name declaration (-N)
2. Computational requirements of the job (-l)
3. Output file or directory for the job (-o)
4. Error file or directory for the job (-e)

It is noteworthy that once a job is submitted it is handled as if the user just logged into the node and thus starts at the home directory. There are a number of standard environment variables automatically assigned by PBS that make it easier to write a bash file the most used of which would be:

1. PBS_O_PATH returns the value of the PATH variable.
2. PBS_O_HOST returns the name of the host on which the qsub command is running.
3. PBS_O_WORKDIR return the absolute path of the current working directory of the qsub command.

The following bash script is an example of a job to be submitted:

```
#!/bin/sh

#PBS -N Name_of_Job
#PBS -e error_file.txt
#PBS -o output_file.txt

cd $PBS_O_WORKDIR
```

```
8  SERVER=$PBS_O_HOST
9
10 echo server is $SERVER
11 echo working is $PBS_O_WORKDIR
12
13 touch one
14 touch two
15 touch three
16 python script.py
17 g++ script.C -o executable
```

Line 1 is a declaration that this is a bash script and is standard bash scripting practice. Declaring the name, output and error is done with a *#PBS* preamble, used to indicate PBS directives setting job attributes. Line 7–11 illustrates some use cases for the standard environment variables. Line 10 and 11 will be written to the *output_file.txt* in the directory where the job was submitted instead of in the home directory. Line 13–15 then creates three files in the *#PBS_O_WORKDIR* directory. This job then runs a python script and complies some C code in lines 16 and 17 respectively. These lines are identical to what one would type if they were doing it interactively.

2 Using the High Performance Computers head node as a virtual private network (VPN)

The technique used here is a patch and not a complete solution by any means. This will allow you to forward the HPC port to your browser and thus allow you to access private/internal pages.

This procedure is given under the assumption that you are using Linux or Mac OS. If you are using Windows, please go to this page and follow the instructions under the "Windows" section.

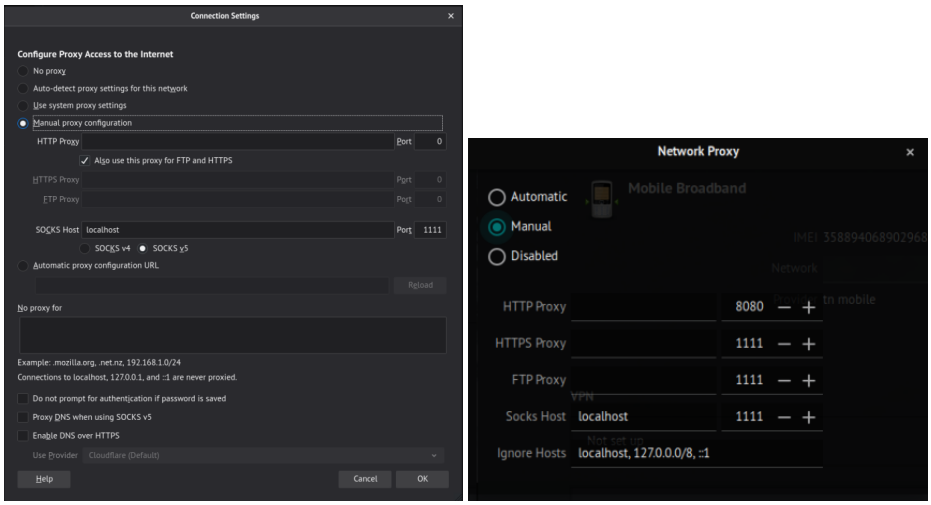
First establish a secure shell connection to the HPC dedicated only for the port forwarding.

```
1 $ ssh -D 1111 -f -C -q -N -p 1510 <username>@uhpc.unam.na
```

A proxy now needs to be setup using this port. In your proxy settings (either of the computer or the browser in use) set up the proxy with the following configuration:

1. SOCKS hostname = localhost
2. port = 1111

Figure: 1a shows the configuration settings for the browser (firefox) and figure: 1b shows the proxy configuration for the computer. Choose any ONE of these methods after the ssh.



(a) Browser configuration (b) Local machine configuration

Figure 1

2.1 H.E.S.S. Users

The above procedure is fine for accessing the H.E.S.S. internal pages. However to access the Namibian WebSummary (base.hess-experiment.eu) you may have to specify the port (:8005).

3 Using Anaconda

The HPC already has a public anaconda installation that can be used by everyone. All that is needed is to initialize conda. Copy these content into a file called, e.g. conda.sh

```
export PATH=$PATH:/home/software/hess/anaconda3/bin:/home/software/hess/
   anaconda3/condabin  #access to anaconda
conda init bash   #For environment activation
conda config --set auto_activate_base false #Do not activate by default
```

Then run it with:

```
$ source conda.sh
```

Log out and back in. Now the anaconda installation can be used without repeating this procedure. There are a few environments already installed such as pyroot activated with:

```
$ conda activate pyroot
```

This pyroot environment uses ROOT version 6.20.4, as a result the proprietary HESS software will not work in this environment. It is highly advised to use

environments when working doing anything on the HPC.
Search available environments using:

```
$ conda env list
```

Create new environments tailored to your needs and activate it as follows:

```
conda create --name <environment name> python=<python version number>
conda activate <environment name>
```

Install packages using:

```
conda install <package>
```

For more, consult the conda documentation.

4 H.E.S.S. Analysis

4.1 Paris Analysis software setup

Before proceeding to the setup below make sure you have completed the first part of section: 3. You must have a working conda environment to get started.

To use the Paris Analysis (PA) software, an environment must be activated. Copy the following lines in a file and run it, e.g. setHESS.sh.

```
## remove gcc v 7 and python 3
export PATH=/home/software/hess/anaconda3/condabin:/opt/ohpc/pub/mpi/openmpi3-gnu7/3.1.0/bin:/opt/ohpc/pub/utils/prun/1.2:/opt/ohpc/pub/utils/autotools/bin:/opt/ohpc/pub/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/opt/pbs/bin
export LD_LIBRARY_PATH=/opt/ohpc/pub/mpi/openmpi3-gnu7/3.1.0/lib
## set up root
source /home/software/hess/root/root_v5.34.36/bin/thisroot.sh
## astroroot
export ISDC_ENV=/home/software/hess/root/astroroot
source $ISDC_ENV/bin/astroroot_init_env.sh
## hess
export HESSROOT=/home/software/hess/ParisAnalysis/HESS_Soft_0-8-34
export PATH=$HESSROOT/bin:$PATH
export LD_LIBRARY_PATH=$HESSROOT/lib:$LD_LIBRARY_PATH
##DATA
export DATA=/home/software/hess/data_temp/DST/Data/Model/Model_HESSII_Hybrid_Prod10_PreProd1
## HESS gnome
export PYTHONPATH=/usr/lib64/python2.7/site-packages:/home/software/hess/root/other_builds/root/lib
conda activate /home/software/hess/anaconda3/envs/py27
alias gpa="python $HESSROOT/parisanalysis/gui/GUI.py &"
alias grq="python $HESSROOT/parisrunquality/gui/GUI.py &"
alias root="TZ=UTC root -l"
export PATH=$HESSROOT/scons/local:$PATH
alias scons="scons.py"
export SCONSFLAGS="--jobs=10"
```

This will not work if the conda is not set up.

```
source setHESS.sh
```

This will activate the Paris Analysis version 34. To activate version 32 change line in 6 of "setHESS.sh" from 34 to 32.

To start the Paris Analysis GUI use the alias:

```
$ gpa
```

To start the Run Quality Selector use the alias:

```
$ grq
```

Feel free to change these alias' in "setHESS.sh" on lines 10 and 11.

For a tab by tab description of the PA software see the PA documentation. Note that this documentation is only accessible from campus unless you use the technique described in section: 2.

4.1.1 Database Access

To access the database a ".dbtoolsrc" file will be needed in the home directory. It's format is as follows.

```
[astro]
 host=host
 port=port
 user=hess
 password=*****
 database=HESS_DAQ_Namibia

[hess]
 host=host
 port=port
 user=hess
 password=*****
 database=HESS_DAQ_Namibia

[calibration]
 host=host
 port=port
 user=hess
 password=*****
 database=CALIBRATION

[analysis]
 host=host
 port=port
 user=hess
 password=*****
 database=ANALYSIS
```

Contact your supervisor for the details of this file.

4.2 Paths and DSTs

The DST version under the DST tab is either "Model_HESSII_Hybrid_Prod10_PreProd1" or "Model_HESSII_Hybrid_Prod8" depending on the version of your data. To check the availability of your data use the following python script:

```
#Test for available runs locally
#feed runlist as argument

import os
import sys

if len(sys.argv) < 2:
    raise Exception('You must feed in the runlist as an argument and the prod type!')

def read_runs():
    file_string = sys.argv[1]
    file = open(file_string, 'r')
    runlist = []

    for line in file:
        runlist.append(int(line))

    return runlist

def check(runlist):
    try:
        prod = sys.argv[2]
    except IndexError:
        sys.exit('WHAT IS THE PROD NUMBER, PROD10 OR PROD8')
    PATH = 'l'
    if prod.lower() == 'prod10':
        PATH = '/home/software/hess/data_temp/DST/Data/Model/Model_HESSII_Hybrid_Prod10_PreProd1/'
    elif prod.lower() == 'prod8':
        PATH = '/home/software/hess/data_temp/DST/Data/Model/Model_HESSII_Hybrid_Prod8/'

    for run in runlist:
        if len(list(str(run))) == 5:
            name = PATH + 'dst_model_0' + str(run) + '_01.root'
        else:
            name = PATH + 'dst_model_' + str(run) + '_01.root'
        #Now test if it is here
        if os.path.exists(name):
            continue
        else:
            print('Run number %i is not here' %run)

runlist = read_runs()
check(runlist)
```

Use with:

```
python <script.py> <runlist> <DST version>
python script.py runlist.list prod10
```

This will print out the run numbers unavailable locally.

Under the Environment tab the relevant *input* and *output* directories of interest are:

1. DST: /home/software/hess/data_temp/DST
2. Tables (prod10): /home/software/hess/Tables/HESS_Soft_0-8-34_HESSII_Prod10_PreProd1
3. Tables (prod8): /home/software/hess/Tables/
   HESS_Soft_0-8-32_HESSII_Prod8_FineSlice_TimeDpdtInhomogOptEff

4.3 Extracting Run Lists

To extract a runlist use the Run Quality Selector. Figure: 2 shows a screen

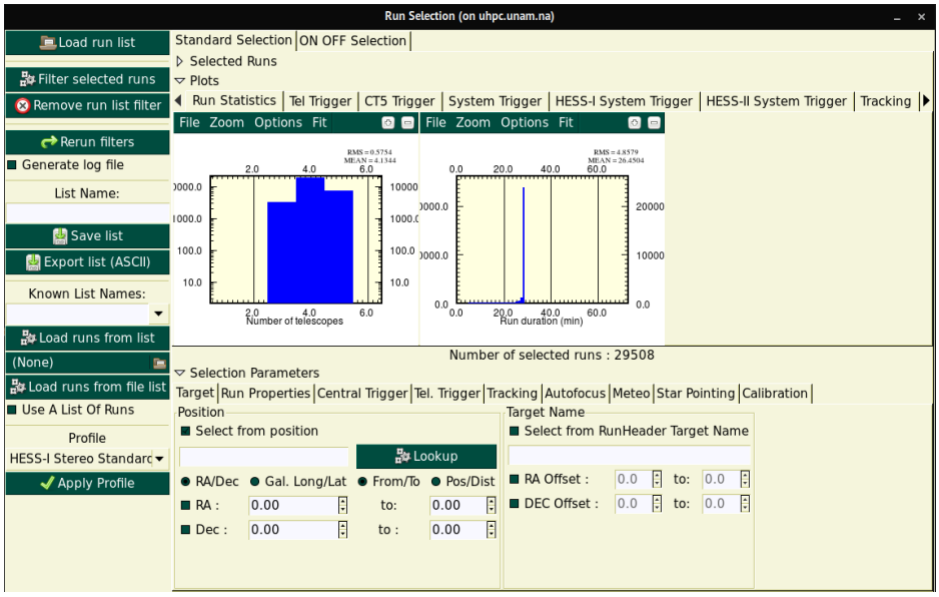


Figure 2: Run Quality Selector

shot of the selector. To extract runs first search for the source in the *Position* parameter as shown in figure: 3 Once the position of the source has been *looked*

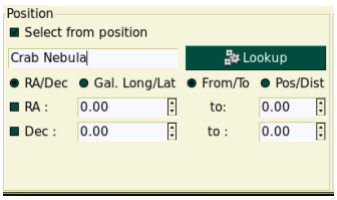


Figure 3: Position Parameter

up, select the *Profile*, i.e. the telescope configuration of the runs you are seeking for.
The runs can now be selected in the *Selected Runs* drop down section and the once selected, write the name of the list in the *List Name* section and Export list as ASCII. Do not Save List. If you change some selection parameters be sure to *Rerun filters* before selecting runs in the drop down section.

4.4 Using H.E.S.S. Python scripts

To be done by Jimmy when he figures out the database issues.