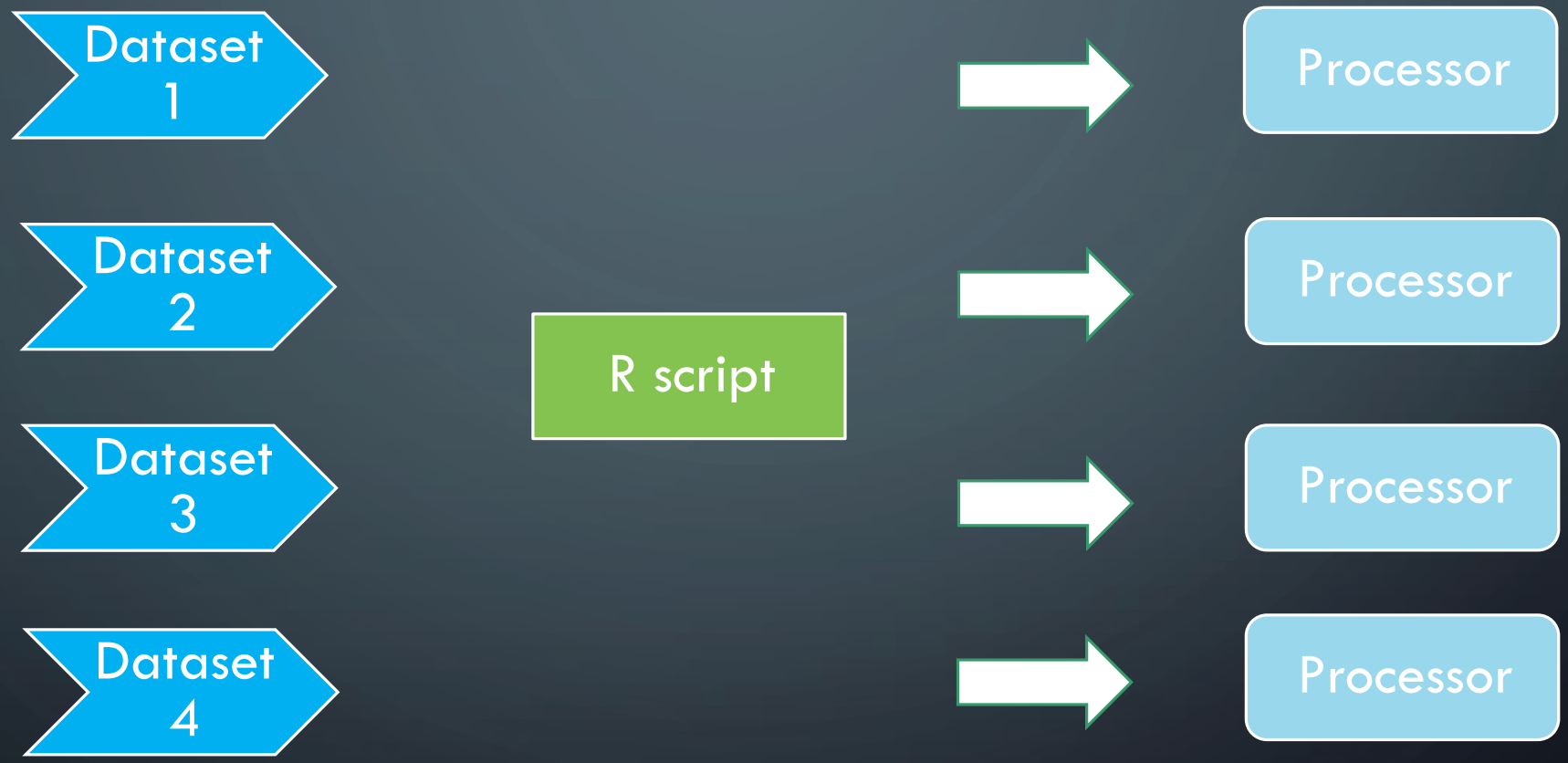
PARALLEL COMPUTING IN R USING WESTGRID CLUSTERS

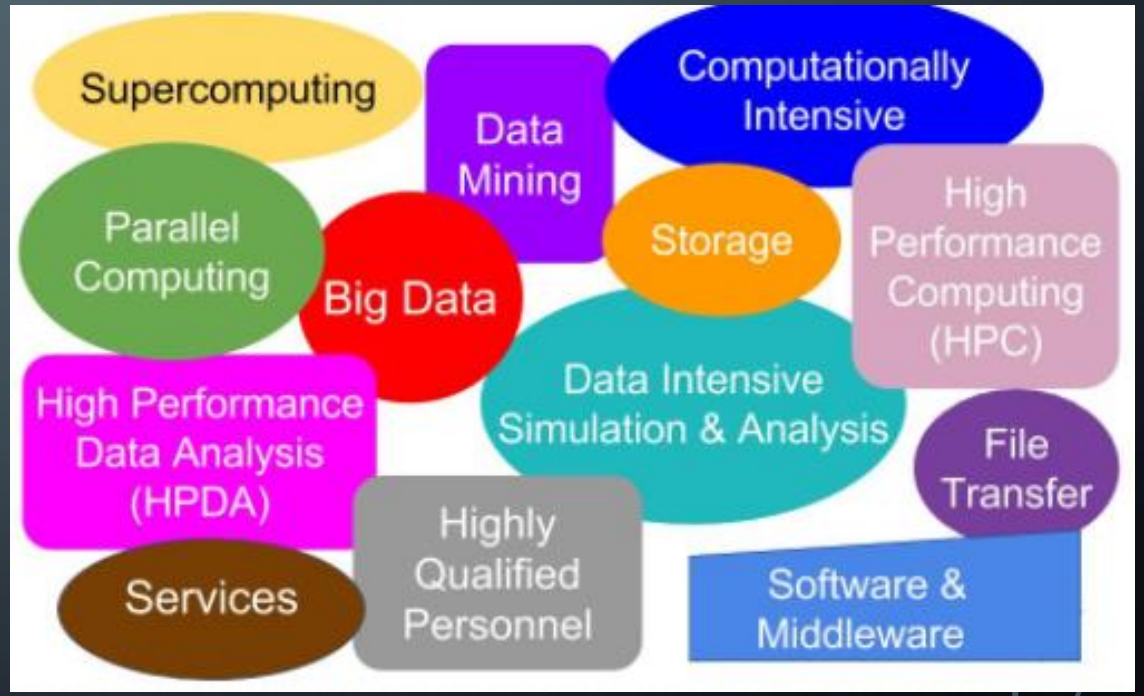
STATGEN GROUP MEETING 10/30/2017

PARALLEL COMPUTING



WHAT IS ADVANCED RESEARCH COMPUTING (ARC)?

- Advanced Research Computing (ARC) is everything beyond a standard desktop workstation
- ARC comes into play when we need more computational power for our research
- This includes
  - Cloud
  - Supercomputers/High Performance Computing (HPC)
  - Data management
  - Data storage
  - Service support
  - Highly qualified personnel
  - etc

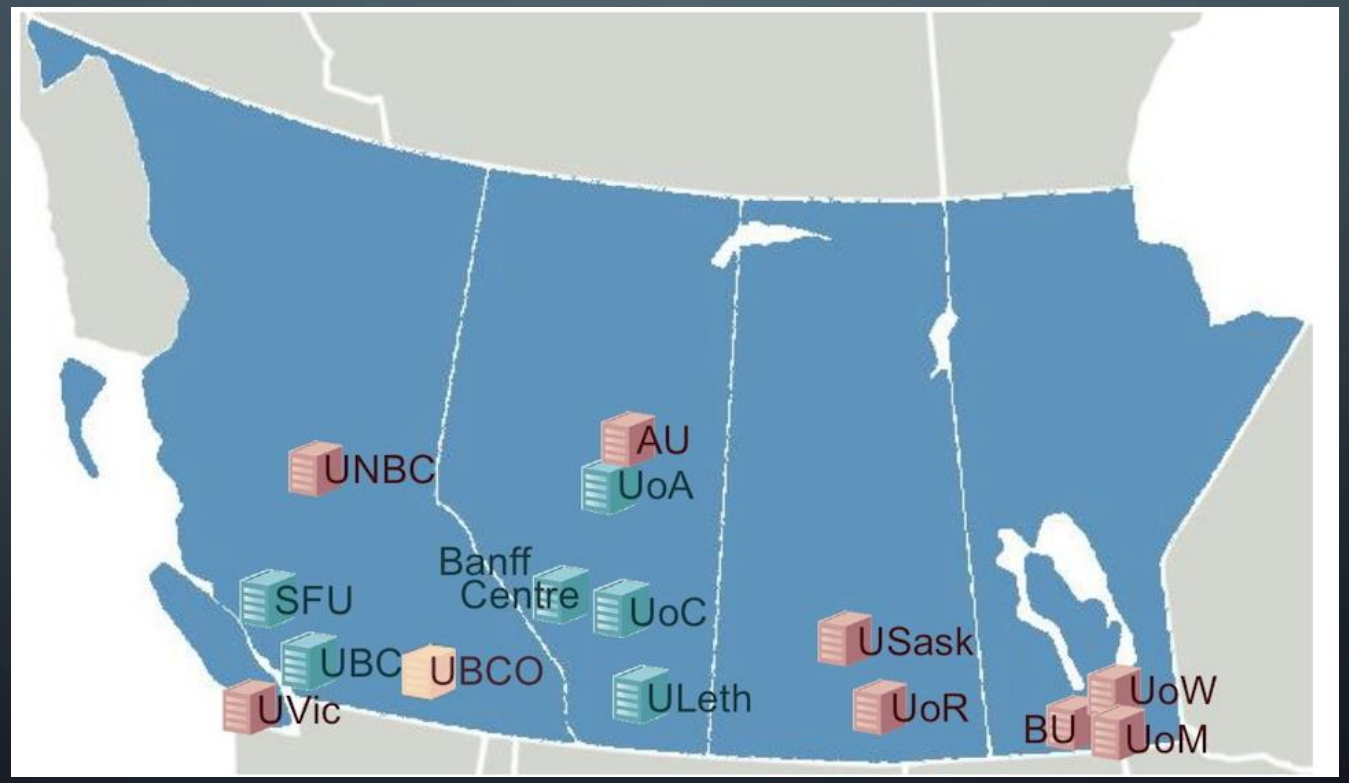


COMPUTE CANADA

- Non-profit umbrella organization
- Provides the essential ARC services and infrastructure for industry and researchers in Canada
- Team of more than 200 experts, employed by 34 partner universities and research institutions across the country
- Regional Consortia

WESTGRID

- Looking at the WestGrid in particular, there are roughly 15 partnerships across four provinces (From BC to MB)



COMPUTATIONAL RESOURCES

System(s)	Site	Cores	Type	Details
Arbutus / Cloud-West / CC-Cloud	University of Victoria	7640	OpenStack Cloud	Visit the CC-Cloud Resources page on the Compute Canada User Wiki for full system details.
Breezy	University of Calgary	384	Shared memory	**NOTE: This system was "defunded" August 31, 2017. Please visit the Migration Process page for more information.** ▪ Appro ▪ 24 nodes: quad-socket, 6-core AMD 2.4GHz nodes ▪ 256 GB per node ▪ Infiniband 4X QDR ▪ Dell FluidFS file system
Bugaboo	Simon Fraser University	4584	Storage, Cluster with fast interconnect	▪ Dell ▪ 160 nodes: 8 cores, Xeon X5430 with 16 GB/node = 1,280 cores (Infiniband, 2:1 blocking) ▪ 254 nodes: 12 cores, Xeon X5650 (212 nodes with 24 GB/node, 32 nodes with 48 GB/node) = 3,048 cores (Infiniband, 2:1 blocking)
Grex	University of Manitoba	3792	Storage, Cluster with fast interconnect	▪ SGI Altix XE 1300 ▪ 316 compute nodes ▪ 2 x 6core Intel Xeon X5650 2.66 MHz processors per node ▪ 24 nodes have 96 GB, 292 nodes have 48 GB ▪ Infiniband 4X QDR

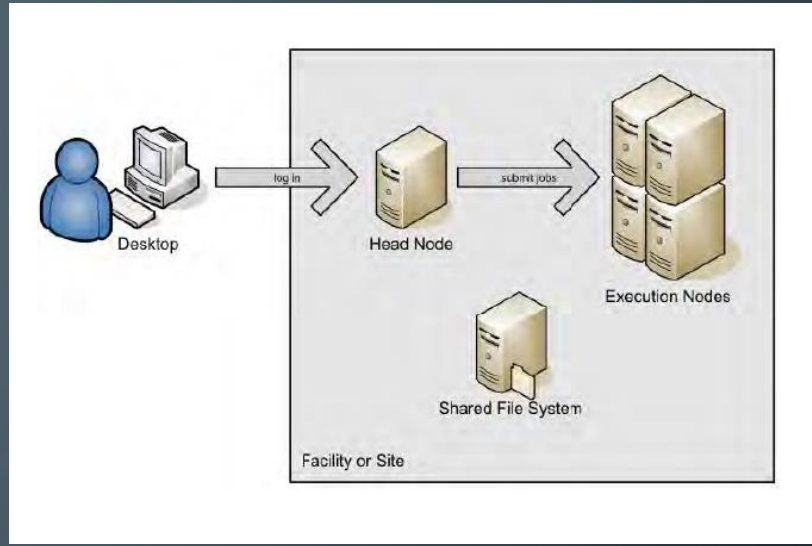
Hungabee	University of Alberta	2048	Shared memory	**NOTE: This system will be "defunded" in Fall 2017. Please visit the Migration Process page for more information.**
				▪ **Special Request Only**
				▪ SGI UV1000, NUMA Shared-memory
				▪ 2048 Intel Xeon E7 cores
				▪ 16 TB total (shared) memory
				▪ NFS: 2 x SGI IS5000 storage arrays
				  ▪ 8 x fibrechannel direct to the UV1000. (short term storage)
				  ▪ 50 TB
				▪ Lustre: 1 x SGI IS16000 array 355 TB. (medium term storage)
				  ▪ Available to BOTH Hungabee and Jasper through QDR IB

COMPUTATIONAL RESOURCES

Jasper	University of Alberta	4160	Cluster with fast interconnect	▪ SGI Altix XE, 400 nodes, 4160 cores and 8320 GB of memory ◦ 204 Xeon X5675 nodes - 12 cores (2 x 6), **24 GB**, 40 Gbit/sec 1:1 Infiniband interconnect ◦ 36 Xeon X5675 nodes - 12 cores (2 x 6), **48 GB**, 40 Gbit/sec 1:1 Infiniband interconnect ◦ 160 Xeon L5420 nodes - 8 cores (2 x 4), **16 GB**, 20 Gbit/sec 2:1 Infiniband interconnect ▪ Lustre parallel distributed filesystem, 356 TB - shared with all nodes via Infiniband
Lattice	University of Calgary	4096	Storage, Cluster with fast interconnect	**NOTE: This system was "defunded" August 31, 2017. Please visit the Migration Process page for more information.** ▪ 512 x 8-core nodes. ▪ Intel Xeon L5520 quad core 2.27 GHz ▪ 12 GB/node ▪ QDR IB (2:1 blocking factor)
Orcinus	University of British Columbia	9600	Storage, Cluster with fast interconnect	▪ Phase 1: 384 nodes, 3072 cores ▪ 8 cores/node ▪ Xeon E5450 3.0GHz ▪ 16 GB Ram ▪ DDR IB ▪ Phase 2: 554 nodes, 6528 cores ▪ 12 cores/node ▪ Xeon X5650 2.66 GHz ▪ QDR IB ▪ IB with 2:1 blocking factor ▪ Phase 1 and Phase 2 share filesystems but otherwise run as separate systems

- More details: https://www.westgrid.ca/support/systems

CONNECTING TO WESTGRID



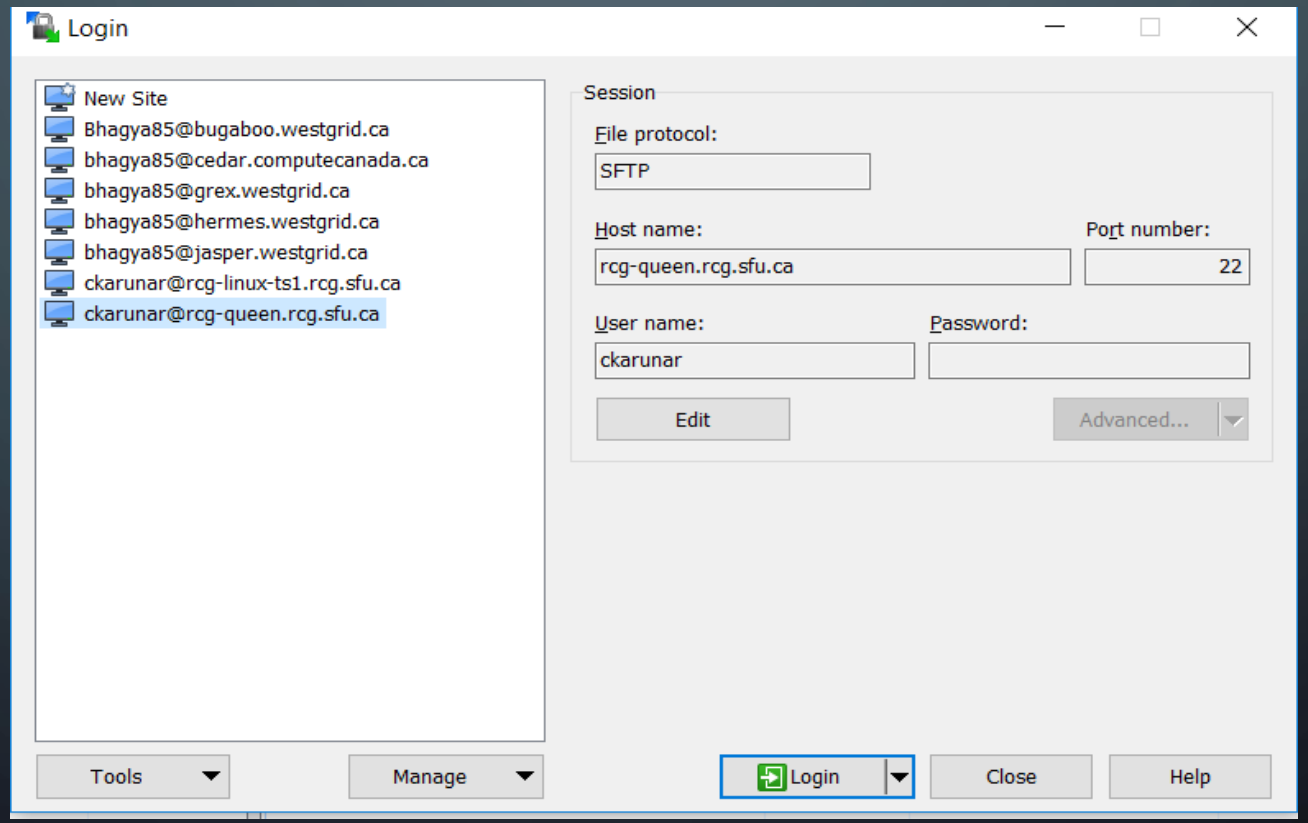
- You have to have a Compute Canada(CC) Account
- Not have a CC account?
  - Try with SFU cluster, Queen
- Since WestGrid consists with cluster system, you will be logging into a head node and from there, you will be submitting jobs to the execution nodes
- Everything you do is a touch oriented which means that you have to make up a script for a job, and these jobs are then submitted and the jobs are put into a queue
- Once jobs are queued, it starts execution based on resource allocation
- To run a job on the HPC cluster, you will need to set up a Portable Batch System (PBS) file
- This PBS file defines the commands and cluster resources used for the job

CONNECTING TO WESTGRID

- Software needs to connect to WestGrid/SFU Queen
- Windows (My focus)
  - Connect with PuTTY (http://www.putty.org/)
  - Connect using software
    - WinSCP (https://winscp.net/eng/download.php)
      - Install PuTTY
      - Install WinSCP
    - MobaXterm (https://mobaxterm.mobatek.net/download.html)
- Mac
  - Connect with terminal
    - ssh –Y myWestGridID@clusterName.westgrid.ca
  - Connect using software
    - Xquartz (https://www.xquartz.org/)

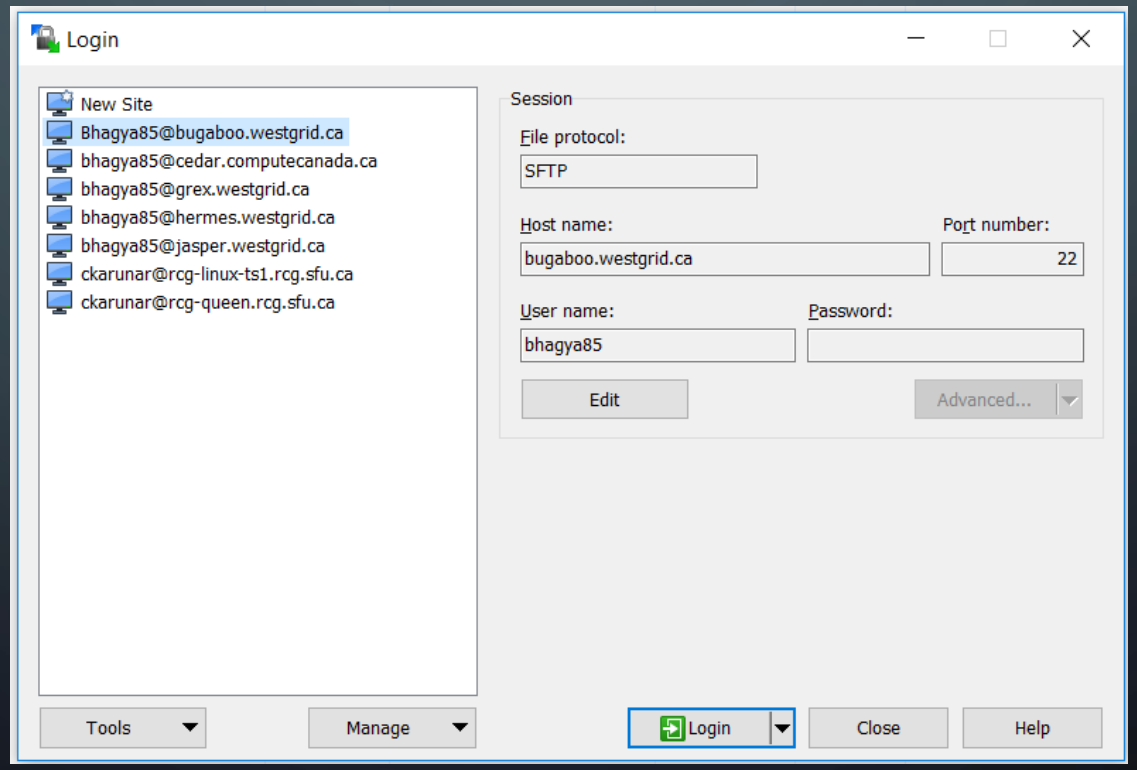
CONNECT USING WINSCP

- SFU Queen (Not a WestGrid cluster)
  - User name and password: your SFU computing ID and the password



CONNECT USING WINSCP

- WestGrid clusters
  - Host name depends on the cluster
    - Ex: For Bugaboo at SFU : bugaboo.westgrid.ca, for Grex at UOM : grex.westgrid.ca
  - User name and password : your WestGrid account username and password



SUBMITTING A JOB: R JOB

- Portable Batch System (PBS) for submitting jobs
- PBS file defines the commands and cluster resources used for the job
- You can write PBS file with the text editor in WinSCP and save it as <file_name>.pbs
- Submit Single R job ?
- Scheduling multiple jobs: R job array ?

SUBMIT A SINGLE R JOB

Step 1 : Write a R script using the text editor in WinSCP and save it as a R file
You need to save your results in a folder as follows

```
x = 1:10
y = 1:10

z = x^2+y^2

save(z, file = "/home/bhagya85/Research/RunJobsEx/test1.Rdata")
```

SUBMIT A SINGLE R JOB

Step 2 : Submit the R job using PBS file

The basic PBS commands to submit a single R job as follows

PBS file: test1.pbs

/home/bhagya85/Research/RunJobsEx/test1.pbs - Bha

```
#PBS -S /bin/bash
#PBS -l procs=1
#PBS -l pmem=1000mb
#PBS -l walltime=00:10:00
#PBS -m bea
#PBS -M ckarunar@sfu.ca

module load R/3.3.1

cd $PBS_O_WORKDIR

Rscript test1.R
```

Description of each command

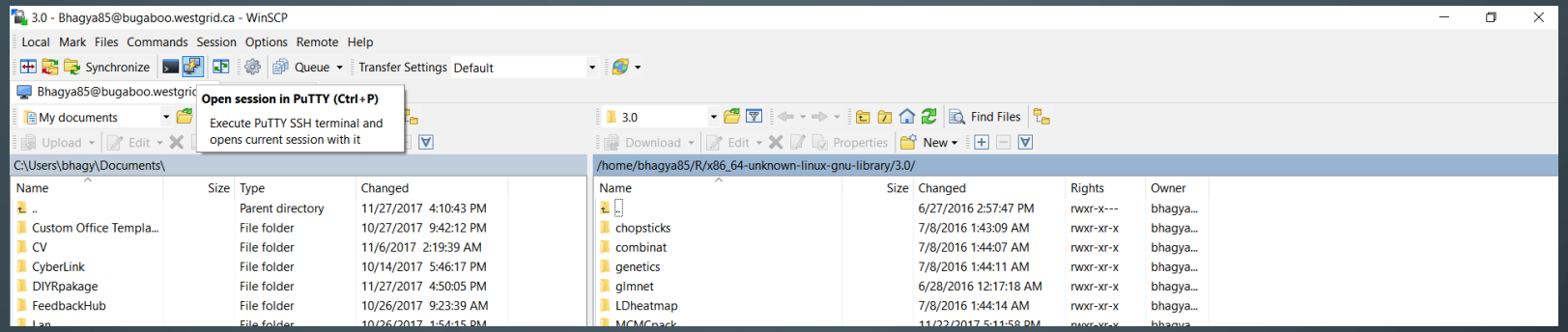
#PBS -S /bin/bash	Sets the shell that the job will be executed on the compute node
#PBS -l nodes=1:ppn=1 #PBS -l procs=1	Requests for 1 processors on 1 node.
#PBS -l walltime=5:00:00	Sets the maximum runtime of 5 hours for your job
#PBS –M <email>	Sets the email address for sending notifications about your job state.
#PBS -m abe	Sets the scedualing system to send you email when a mail is sent when the job is aborted by the batch system. b mail is sent when the job begins execution. e mail is sent when the job terminates.

More about PBS commands : https://www.westgrid.ca/files/PBS%20Script_0.pdf

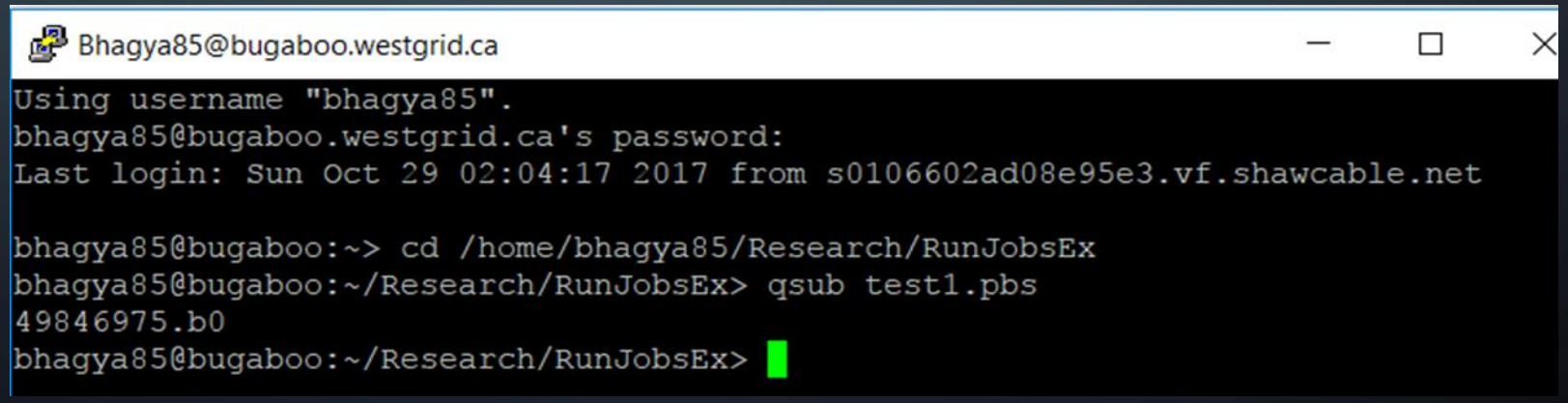
SUBMIT A SINGLE R JOB

Step3: Submit PBS file

1. In WinSCP, open PuTTY session



2. Then in PuTTY session, change the directory to your working directory and submit the PBS file using 'qsub' command as follows



Job ID →

SUBMIT SINGLE R JOB

Step4: Check job status

Command	What its used for
jobinfo -j	List all your jobs and their state
qstat –t –u $USER	List all your array jobs and the subcomponents and their state.
qstat –a	List all jobs on the system and their state.
qstat –r	List all running jobs on the system.
showq	List all jobs on the system and their state.
showq –i	List all jobs being considered for scheduling and their priority
showq –b	Lists all blocked (unable to be run) jobs
qstat –f <Jobid>	List detailed information on Job
checkjob <Jobid>	List detailed information on Job
checkjob -v -v <Jobid>	List detailed information on Job, including history and why it is not running now on each node.

Ex: Checking job status : qstat –t –u <westgrid_username>

```
bhagya85@bugaboo:~> cd /home/bhagya85/Research/RunJobsEx
bhagya85@bugaboo:~/Research/RunJobsEx> qsub test1.pbs
50374505.b0
bhagya85@bugaboo:~/Research/RunJobsEx> qstat -t -u bhagya85

b0:
                                                                  Req'd       Req'd       Elap
Job ID                  Username    Queue    Jobname          SessID  NDS   TSK   Memory      Time    S   Time
----------------------- ----------- -------- ---------------- ------ ----- ------ --------- --------- - ---------
50374505.b0             bhagya85    q1       test1.pbs           --    --       1    1000mb  00:10:00 Q       --
bhagya85@bugaboo:~/Research/RunJobsEx>
```

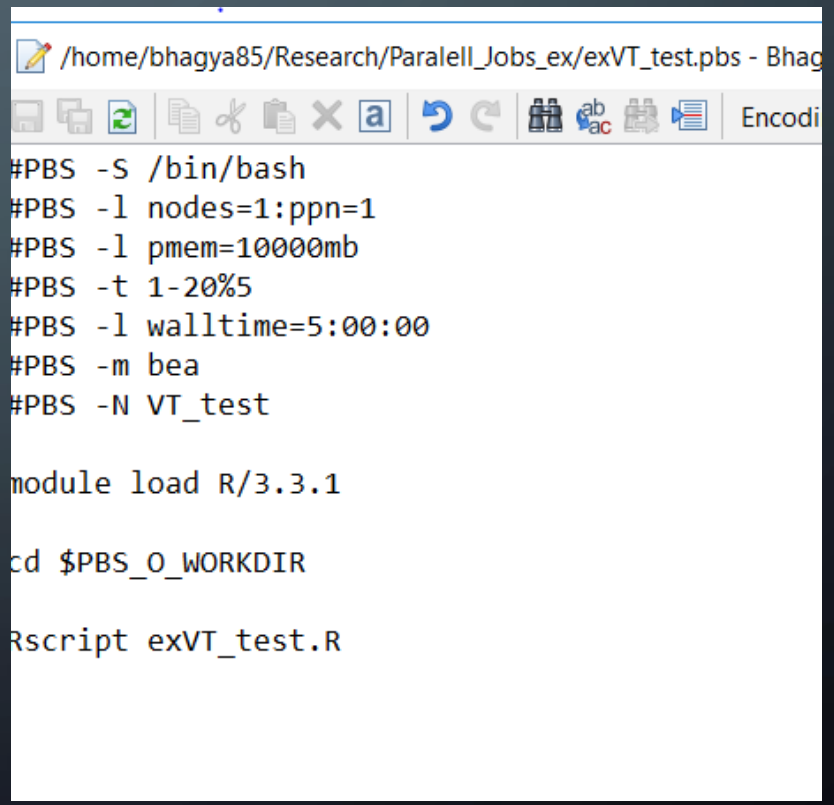
SUBMIT A R JOB ARRAY

- Job arrays in PBS are an easy way to submit multiple similar jobs
- The only difference in them is the array index in PBS file
- You can use the array index in your PBS script to run each task with a different set of parameters, load different data files, or any other operation that requires an unique index
- Some small changes in R script

SUBMIT A R JOB ARRAY: PBS FILE

- Ask for a job array in one of the following ways:
  - #PBS –t 1-100 job array100 jobs numbered 1-100
  - #PBS –t 1,2,3,5,7 job array with 5 jobs with indexes [1,2,3,5,7]
  - #PBS –t 1-100%5 job array 100 jobs numbered 1-100 with a maximum of 5 running at any time
- Ex: PBS script that asks for an array job. Each running the same R script, exVT_test.R

Request 20 array jobs, run five at a time



SUBMIT A R JOB ARRAY: R SCRIPT

- When you submit the job array, the R script has to be written as a R function
  - Ex: analyseDat() of R script, exVT_test.R
- Each running the same script, with the individual jobs identified by a "PBS_ARRAYID" variable.
- The PBS_ARRAYID is implemented as a Unix shell environment variable that is set on each shell running an individual job.
- R script that analyses one dataset taking a dataset ID as input. The dataset ID will be read in from the PBS_ARRAYID environment variable set by the cluster.
- The following Rscript runs a R function called 'Vtscan()' inside the 'analyseDat()' through 200 datasets and save the result for each dataset separately

/home/bhagya85/Research/Paralell_Jobs_ex/exVT_test.R - Bhagya85@bugaboo.westgrid.ca - Editor - WinSCP

Encoding Color

```
# Define a function that takes a dataset ID as input.
analyseDat = function(datasetID) {
  # suppose we read in data from files (ex: .Rdata) .RdataX for X=1,...,200

  # Call the Rscript that includes function 'VTscan()'
  source("/home/bhagya85/Research/Paralell_Jobs_ex/vtscan.R")

  # Read the .Rdata file
  dat = load(sprintf("/home/bhagya85/Research/Paralell_Jobs_ex/testMat%d.Rdata", datasetID))
  # do the analyses
  out = VTscan(dat) # my R function. This should be a R function.
  # save results
  save(paste0("outfile",datasetID,".Rdata"))
}
# Now call analyseDat on a datasetID that will be read from
# Unix environment variable "PBS_ARRAYID", which is created by
# the cluster for each incarnation of the job.
dID = Sys.getenv("PBS_ARRAYID") # PBS_ARRAYID specified in 'exVT.pbs'
analyseDat(dID)
```

CHECK R JOB ARRAY STATUS

- There is a naming convention for jobs in array: Job array with 3 jobs: jobname[1], jobname[2], jobname[3]
- Check the array job status using ‘qstat’ command

Bhagya85@bugaboo.westgrid.ca

```
bhagya85@bugaboo:~/Research/RunJobsEx/ParallelJobs> qsub exVT_test.pbs
50376708[].b0
bhagya85@bugaboo:~/Research/RunJobsEx/ParallelJobs> qstat -t -u bhagya85

b0:
                                                                  Req'd       Req'd       Elap
Job ID                  Username    Queue    Jobname          SessID  NDS   TSK   Memory      Time    S   Time
----------------------- ----------- -------- ---------------- ------ ----- ------ --------- --------- - ---------
50376708[1].b0          bhagya85    q1       VT_test-1           --      1      1   10000mb  05:00:00 Q       --
50376708[2].b0          bhagya85    q1       VT_test-2           --      1      1   10000mb  05:00:00 Q       --
50376708[3].b0          bhagya85    q1       VT_test-3           --      1      1   10000mb  05:00:00 Q       --
50376708[4].b0          bhagya85    q1       VT_test-4           --      1      1   10000mb  05:00:00 Q       --
50376708[5].b0          bhagya85    q1       VT_test-5           --      1      1   10000mb  05:00:00 Q       --
50376708[6].b0          bhagya85    q1       VT_test-6           --      1      1   10000mb  05:00:00 H       --
50376708[7].b0          bhagya85    q1       VT_test-7           --      1      1   10000mb  05:00:00 H       --
50376708[8].b0          bhagya85    q1       VT_test-8           --      1      1   10000mb  05:00:00 H       --
50376708[9].b0          bhagya85    q1       VT_test-9           --      1      1   10000mb  05:00:00 H       --
50376708[10].b0         bhagya85    q1       VT_test-10          --      1      1   10000mb  05:00:00 H       --
50376708[11].b0         bhagya85    q1       VT_test-11          --      1      1   10000mb  05:00:00 H       --
50376708[12].b0         bhagya85    q1       VT_test-12          --      1      1   10000mb  05:00:00 H       --
50376708[13].b0         bhagya85    q1       VT_test-13          --      1      1   10000mb  05:00:00 H       --
50376708[14].b0         bhagya85    q1       VT_test-14          --      1      1   10000mb  05:00:00 H       --
50376708[15].b0         bhagya85    q1       VT_test-15          --      1      1   10000mb  05:00:00 H       --
50376708[16].b0         bhagya85    q1       VT_test-16          --      1      1   10000mb  05:00:00 H       --
50376708[17].b0         bhagya85    q1       VT_test-17          --      1      1   10000mb  05:00:00 H       --
50376708[18].b0         bhagya85    q1       VT_test-18          --      1      1   10000mb  05:00:00 H       --
50376708[19].b0         bhagya85    q1       VT_test-19          --      1      1   10000mb  05:00:00 H       --
50376708[20].b0         bhagya85    q1       VT_test-20          --      1      1   10000mb  05:00:00 H       --
bhagya85@bugaboo:~/Research/RunJobsEx/ParallelJobs>
```

THANK YOU