Alphafold2(v2.2) on NUS HPC GPU Cluster

By Ku Wee Kiat
AI Engineer/HPC Specialist
Research Computing, NUS IT

Content

- Register for NUS HPC Account
- Accessing the HPC Environment
  - Upload/Downloading Data
- Resources
- Running Alphafold2 on NUS HPC GPU Cluster
- The PBS Job Scheduler
  - Submitting a Job
  - Checking Jobs
- From Start to End

Register for NUS HPC Account

Go to:
https://nusit.nus.edu.sg/services/hpc/getting-started-hpc/register-for-hpc/

and fill in the necessary information.

Guide: https://nusit.nus.edu.sg/services/getting-started/registration-guide/

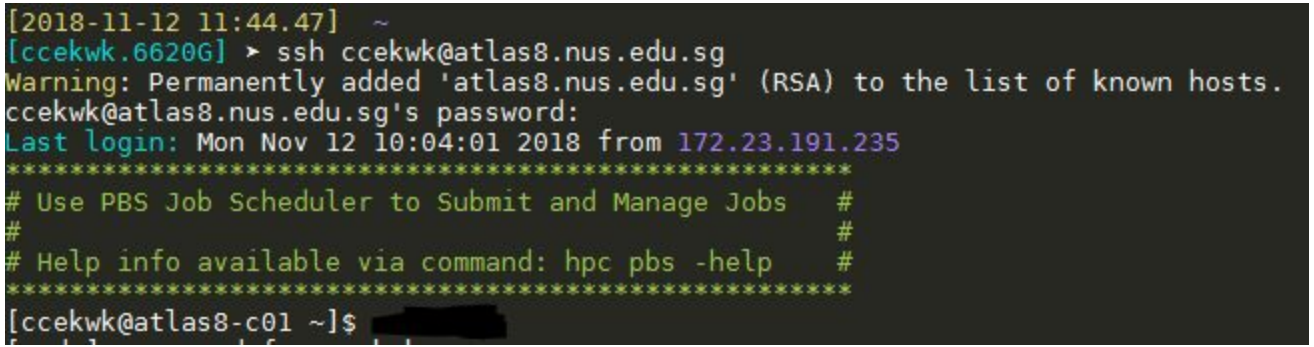
Access

Access

- Login via ssh to NUS HPC login nodes
  - atlas9
- If you are connecting from outside NUS network, please connect to VPN first
  - http://webvpn.nus.edu.sg

Access

OS	Access Method	Command
Linux	ssh from terminal	`ssh nusnet_id@atlas9-c01.nus.edu.sg`
MacOS	ssh from terminal	`ssh username@hostname`
Windows	ssh using mobaxterm or putty or terminal (**powershell**)	`ssh username@hostname`


[2018-11-12 11:44.47] ~
[ccekwk.6620G] ➤ ssh ccekwk@atlas8.nus.edu.sg
Warning: Permanently added 'atlas8.nus.edu.sg' (RSA) to the list of known hosts.
ccekwk@atlas8.nus.edu.sg's password:
Last login: Mon Nov 12 10:04:01 2018 from 172.23.191.235
****
Use PBS Job Scheduler to Submit and Manage Jobs
#
Help info available via command: hpc pbs -help
****
[ccekwk@atlas8-c01 ~]$

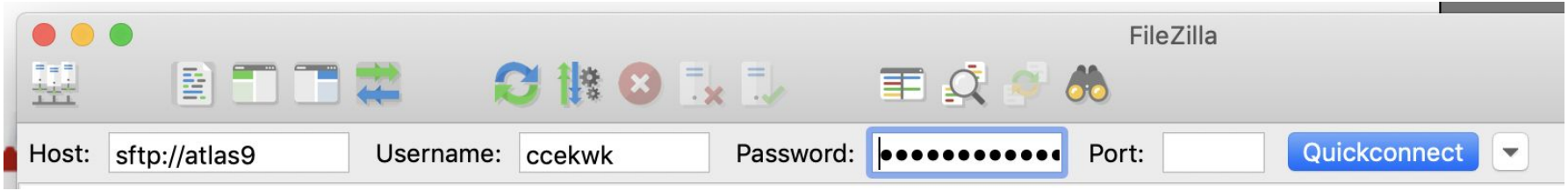
Logging in

File Transfer

1. Mobaxterm built-in sftp client
2. **Filezilla client**
3. Linux/Mac OS/Windows Terminal Tools
    a. scp
    b. rsync
    c. sftp

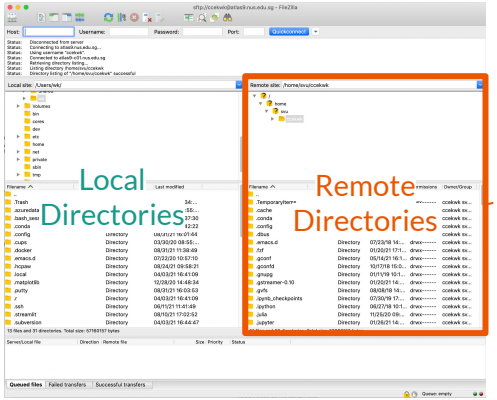
Filezilla

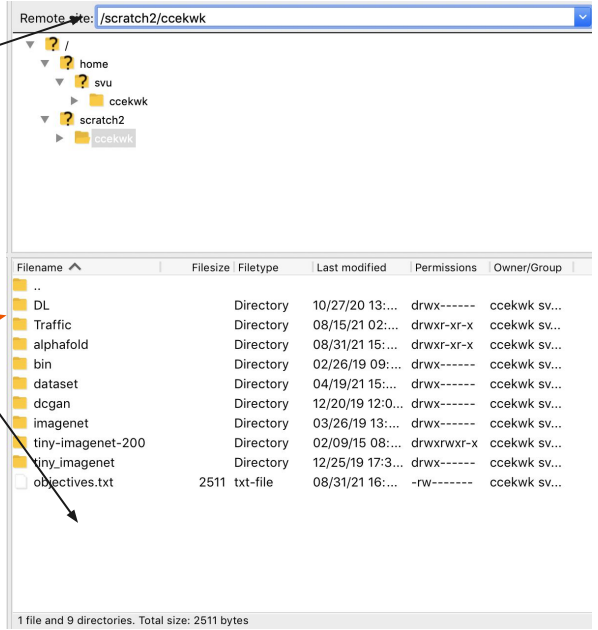
- Download the Filezilla client for your computer's operating system: https://filezilla-project.org/download.php
- Log in
  - Host: sftp://atlas9.nus.edu.sg
  - Username: Your NUSNET ID
  - Password: Your NUSNET Password
- When prompted to "Trust this host" -> Click OK



Uploading a File

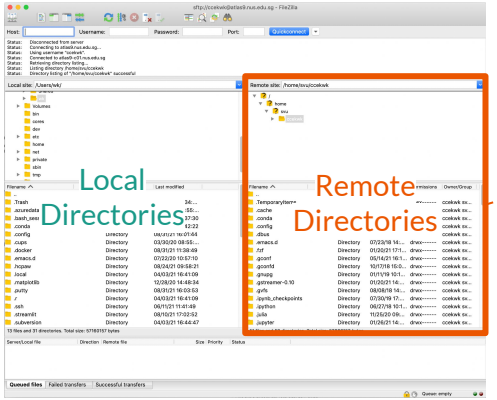
- On the center-right panel, enter the path to your working directory in Remote Site box and hit Enter
- Drag and Drop files you want to upload here

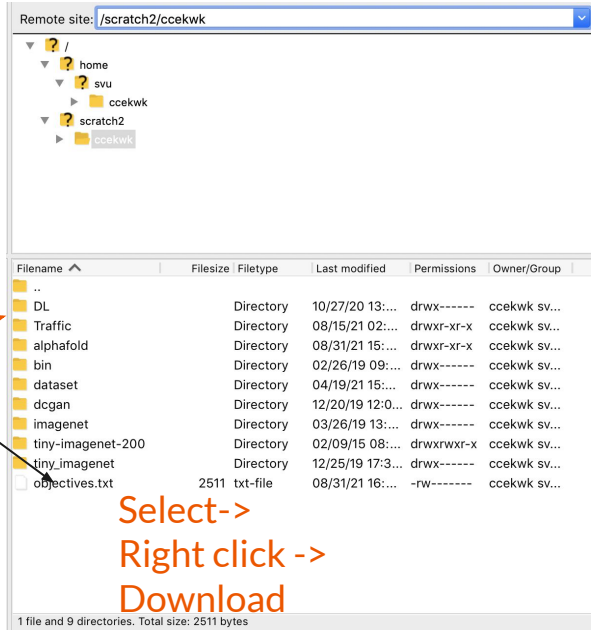




Downloads

To download a folder or files, just select them and right click -> Download





Select->
Right click ->
Download

Resources

Resources: Hardware

GPU Clusters

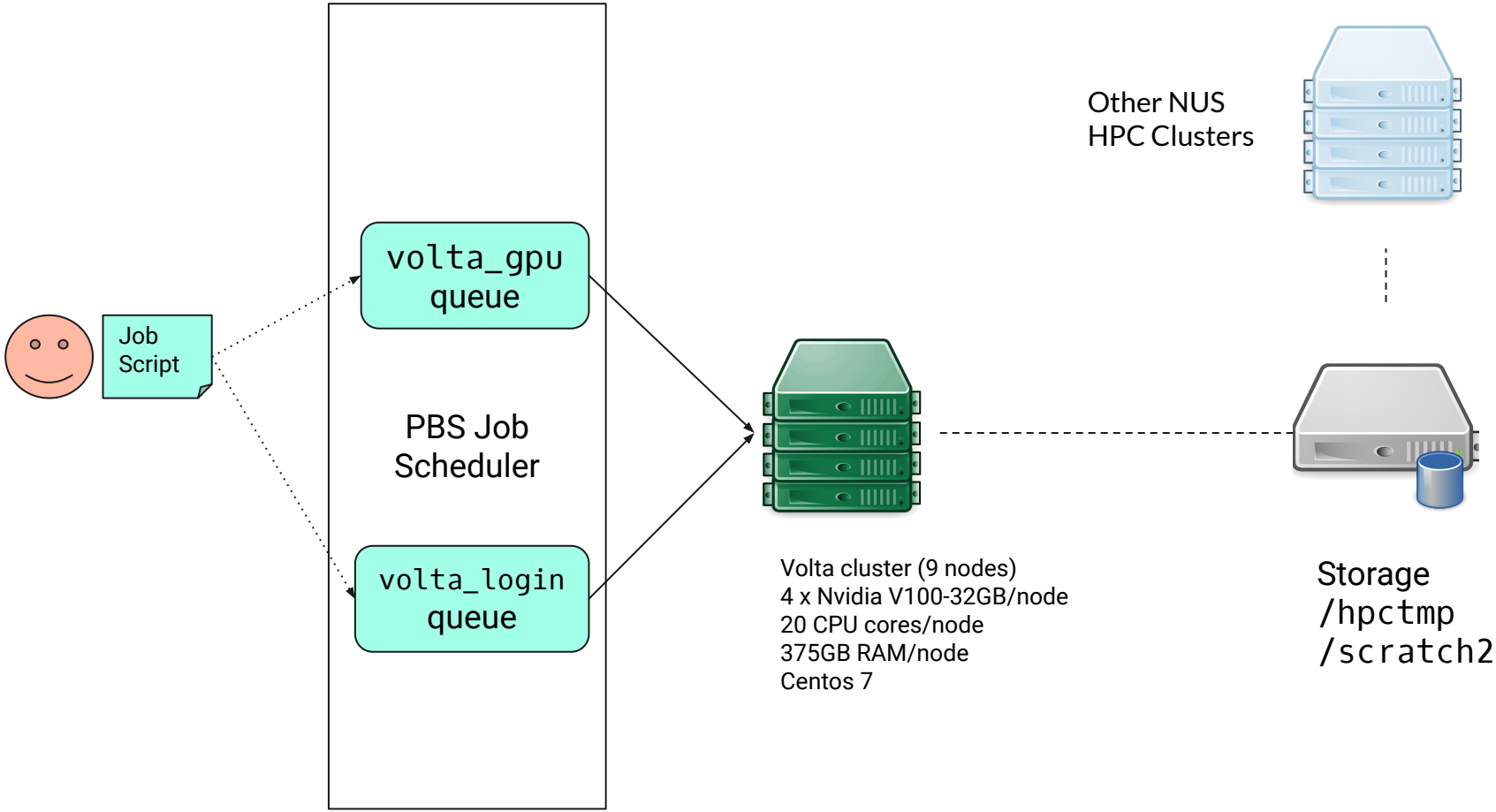
- 9 nodes x 4 Nvidia Tesla V100-32GB

No internet access on Volta Servers

Resources: Hardware/Storage

Directories	Feature	Disk Quota	Backup	Description
`/home/svu/$USERID`	Global	20 GB	Snapshot	Home Directory. U:drive on your PC.
`/hpctmp/$USERID`	Local on All Atlas/Volta cluster	500 GB	No	Working Directory. Files older than 60 days are purged automatically
`/scratch/$USERID`	Local to each Volta node	5 TB	No	For quick read/write access to datasets. Create a folder with your NUSNET ID. Routinely purged.
`/scratch2/$USERID`	Available on Atlas 9 and Volta Cluster	1 TB	No	For quick read/write access to datasets. Create a folder with your NUSNET ID. Routinely purged.

Note: Type “hpc s” to check your disk quota for your home directory


Job
Script
volta_gpu
queue
PBS Job
Scheduler
volta_login
queue
Other NUS
HPC Clusters
Volta cluster (9 nodes)
4 x Nvidia V100-32GB/node
20 CPU cores/node
375GB RAM/node
Centos 7
Storage
/hpctmp
/scratch2

Batch Jobs

Queue Resources

Max RAM = 142gb
Max No. of CPU cores = 20
Max No. of GPUs = 2
Max Walltime = 72:00:00
Minimum No. of CPU cores = 5
Minimum No. of GPU = 1
Default Walltime = 04:00:00
Request CPU Core in increments of 1

Modes

- Monomer - Full DBs
- Monomer - Reduced DBs
- Multimer - Full DBs
- Multimer - Reduced DBs

Sample Job Script (Monomer)

For Alphafold2 **Monomer** Batch Jobs

Note:

1. **Do not copy and paste** the job script in the next slide directly into your text editor.
2. Please **type it out manually** to avoid hidden characters.
3. Ensure that the jobscript **conforms to BASH syntax.**

```
#!/bin/bash

#PBS -P alphafold_project_name
#PBS -j oe
#PBS -N alphafold_job_name
#PBS -q volta_gpu
#PBS -l select=1:ncpus=10:mem=100gb:ngpus=1
#PBS -l walltime=15:00:00

cd $PBS_O_WORKDIR;
np=$(cat ${PBS_NODEFILE} | wc -l);

##------ THE ONLY PART FOR YOU TO CHANGE ------
## User settings
INPUT_FASTA_FILE_PATH=wcrC_39.fasta;  ## "my_abc123.fasta" is your input *.fasta file.
OUTPUT_DIR=`pwd`/alphafold22_output_1;       ## "alphafold_output_1" defines output folder name.

MAX_TEMPLATE_DATE='2022-03-30' # yyyy-mm-dd format
MULTIMER_PREDICTIONS_PER_MODEL=5
DB_PRESET=full_dbs     # db_presets: full_dbs, reduced_dbs
MODEL_PRESET='monomer' # model_preset: monomer, monomer_casp14, monomer_ptm, multimer

# Create output directory
mkdir -p ${OUTPUT_DIR}

##------  END TO CHANGE ------------------------

## Might not need to change until there is a newer version ##
## Change this to a newer alphafold container when it is released
IMAGE=/app1/common/singularity-img/3.0.0/alphafold/alphafold_v2.2.sif
## END ##

## DO NOT CHANGE ##
# Params and DB paths
ALPHAFOLD_DATA_PATH=/scratch2/biodata/alphafold/database/
ALPHAFOLD_DATA_PATH2=/scratch2/biodata/alphafold/database_v2_2/
ALPHAFOLD_MODELS=/scratch2/biodata/alphafold/database_v2_2/params

## Do not remove ##
mkdir -p alphafold/common
cp -n /scratch2/biodata/alphafold/setup_files/stereo_chemical_props.txt alphafold/common/
```

Orange is user configurable
Green is updatable
Black is fixed

Path to your fasta file.
No spaces allowed

Path to your desired output folder.
No spaces allowed.

remove to not set a limit on PDB template date or just date limit as needed

Continued on next slide

Orange is user configurable
Green is updatable
Black is fixed

```
##--- To start & run Alphafold2 in Singularity container. ----
singularity run --nv \
 -B $ALPHAFOLD_DATA_PATH:/data \
 -B $ALPHAFOLD_DATA_PATH2:/data2 \
 -B $ALPHAFOLD_MODELS \
 -B .:/etc \
 --pwd `pwd` $IMAGE  \
 --data_dir=/data \
 --output_dir=$OUTPUT_DIR  \
 --fasta_paths=$INPUT_FASTA_FILE_PATH  \
 --uniref90_database_path=/data/uniref90/uniref90.fasta  \
 --mgnify_database_path=/data/mgnify/mgy_clusters.fa   \
 --bfd_database_path=/data/bfd/bfd_metaclust_clu_complete_id30_c90_final_seq.sorted_opt \
 --uniclust30_database_path=/data/uniclust30/uniclust30_2018_08/uniclust30_2018_08 \
 --template_mmcif_dir=/data2/pdb_mmcif/mmcif_files  \
 --pdb70_database_path=/data/pdb70/pdb70  \
 --obsolete_pdbs_path=/data2/pdb_mmcif/obsolete.dat \
 --num_multimer_predictions_per_model=$MULTIMER_PREDICTIONS_PER_MODEL \
 --model_preset=$MODEL_PRESET \
 --max_template_date=$MAX_TEMPLATE_DATE \
 --run_relax=True \
 --use_gpu_relax=True \
 --db_preset=$DB_PRESET > stdout.$PBS_JOBID 2> stderr.$PBS_JOBID
```

Sample Job Script (Monomer Reduced DBs)

For Alphafold2 **Monomer Reduced DBs** Batch Jobs

Note:

1. **Do not copy and paste** the job script in the next slide directly into your text editor.
2. Please **type it out manually** to avoid hidden characters.
3. Ensure that the jobscript **conforms to BASH syntax.**

```
#!/bin/bash

#PBS -P alphafold_project_name
#PBS -j oe
#PBS -N alphafold_job_name
#PBS -q volta_gpu
#PBS -l select=1:ncpus=10:mem=100gb:ngpus=1
#PBS -l walltime=15:00:00

cd $PBS_O_WORKDIR;
np=$(cat ${PBS_NODEFILE} | wc -l);

##------ THE ONLY PART FOR YOU TO CHANGE ------
## User settings
INPUT_FASTA_FILE_PATH=wcrC_39.fasta;  ## "my_abc123.fasta" is your input *.fasta file.
OUTPUT_DIR=`pwd`/alphafold22_output_1;       ## "alphafold_output_1" defines output folder name.

MAX_TEMPLATE_DATE='2022-03-30' # yyyy-mm-dd format
MULTIMER_PREDICTIONS_PER_MODEL=5
DB_PRESET=reduced_dbs  # db_presets: full_dbs, reduced_dbs
MODEL_PRESET='monomer' # model_preset: monomer, monomer_casp14, monomer_ptm, multimer

# Create output directory
mkdir -p ${OUTPUT_DIR}

##------  END TO CHANGE ------------------------

## Might not need to change until there is a newer version ##
## Change this to a newer alphafold container when it is released
IMAGE=/app1/common/singularity-img/3.0.0/alphafold/alphafold_v2.2.sif
## END ##

## DO NOT CHANGE ##
# Params and DB paths
ALPHAFOLD_DATA_PATH=/scratch2/biodata/alphafold/database/
ALPHAFOLD_DATA_PATH2=/scratch2/biodata/alphafold/database_v2_2/
ALPHAFOLD_MODELS=/scratch2/biodata/alphafold/database_v2_2/params

## Do not remove ##
mkdir -p alphafold/common
cp -n /scratch2/biodata/alphafold/setup_files/stereo_chemical_props.txt alphafold/common/
```

Orange is user configurable
Green is updatable
Black is fixed

Path to your fasta file.
No spaces allowed

Path to your desired output folder.
No spaces allowed.

remove to not set a limit on PDB template date or just date limit as needed

Continued on next slide

Orange is user configurable
Green is updatable
Black is fixed

```
##--- To start & run Alphafold2 in Singularity container. ----
singularity run --nv \
 -B $ALPHAFOLD_DATA_PATH:/data \
 -B $ALPHAFOLD_DATA_PATH2:/data2 \
 -B $ALPHAFOLD_MODELS \
 -B .:/etc \
 --pwd `pwd` $IMAGE  \
 --data_dir=/data \
 --output_dir=$OUTPUT_DIR  \
 --fasta_paths=$INPUT_FASTA_FILE_PATH  \
 --uniref90_database_path=/data/uniref90/uniref90.fasta  \
 --mgnify_database_path=/data/mgnify/mgy_clusters.fa   \
 --small_bfd_database_path=/data2/small_bfd/bfd-first_non_consensus_sequences.fasta \
 --template_mmcif_dir=/data2/pdb_mmcif/mmcif_files  \
 --pdb70_database_path=/data/pdb70/pdb70  \
 --obsolete_pdbs_path=/data2/pdb_mmcif/obsolete.dat \
 --num_multimer_predictions_per_model=$MULTIMER_PREDICTIONS_PER_MODEL \
 --model_preset=$MODEL_PRESET \
 --max_template_date=$MAX_TEMPLATE_DATE \
 --run_relax=True \
 --use_gpu_relax=True \
 --db_preset=$DB_PRESET > stdout.$PBS_JOBID 2> stderr.$PBS_JOBID
```

Sample Job Script (Multimer)

For Alphafold2 **Multimer** Batch Jobs

Note:

1. **Do not copy and paste** the job script in the next slide directly into your text editor.
2. Please **type it out manually** to avoid hidden characters.
3. Ensure that the jobscript **conforms to BASH syntax.**

```
#!/bin/bash

#PBS -P alphafold_project_name
#PBS -j oe
#PBS -N alphafold_job_name
#PBS -q volta_gpu
#PBS -l select=1:ncpus=10:mem=100gb:ngpus=1
#PBS -l walltime=20:00:00

cd $PBS_O_WORKDIR;
np=$(cat ${PBS_NODEFILE} | wc -l);

##------ THE ONLY PART FOR YOU TO CHANGE ------
## User settings
INPUT_FASTA_FILE_PATH=P10_trimer.fasta;   ## "my_abc123.fasta" is your input *.fasta file.
OUTPUT_DIR=`pwd`/alphafold22_output_1_m;   ## "alphafold_output_1" defines output folder name.

MAX_TEMPLATE_DATE='2022-03-30' # yyyy-mm-dd format
MULTIMER_PREDICTIONS_PER_MODEL=5
DB_PRESET=full_dbs     # db_presets: full_dbs, reduced_dbs
MODEL_PRESET='multimer' # model_preset: monomer, monomer_casp14, monomer_ptm, multimer

# Create output directory
mkdir -p ${OUTPUT_DIR}

##------  END TO CHANGE ------------------------

## Might not need to change until there is a newer version ##
## Change this to a newer alphafold container when it is released
IMAGE=/app1/common/singularity-img/3.0.0/alphafold/alphafold_v2.2.sif
## END ##

## DO NOT CHANGE ##
# Params and DB paths
ALPHAFOLD_DATA_PATH=/scratch2/biodata/alphafold/database/
ALPHAFOLD_DATA_PATH2=/scratch2/biodata/alphafold/database_v2_2/
ALPHAFOLD_MODELS=/scratch2/biodata/alphafold/database_v2_2/params

## Do not remove ##
mkdir -p alphafold/common
cp -n /scratch2/biodata/alphafold/setup_files/stereo_chemical_props.txt alphafold/common/
```

Orange is user configurable
Green is updatable
Black is fixed

Path to your fasta file.
No spaces allowed

Path to your desired output folder.
No spaces allowed.

Continued on next slide

Orange is user configurable
Green is updatable
Black is fixed

```
##--- To start & run Alphafold2 in Singularity container. ----
singularity run --nv \
 -B $ALPHAFOLD_DATA_PATH:/data \
 -B $ALPHAFOLD_DATA_PATH2:/data2 \
 -B $ALPHAFOLD_MODELS \
 -B .:/etc \
 --pwd `pwd` $IMAGE  \
 --data_dir=/data \
 --output_dir=$OUTPUT_DIR  \
 --fasta_paths=$INPUT_FASTA_FILE_PATH  \
 --uniref90_database_path=/data/uniref90/uniref90.fasta  \
 --mgnify_database_path=/data/mgnify/mgy_clusters.fa   \
 --bfd_database_path=/data/bfd/bfd_metaclust_clu_complete_id30_c90_final_seq.sorted_opt \
 --uniclust30_database_path=/data/uniclust30/uniclust30_2018_08/uniclust30_2018_08 \
 --template_mmcif_dir=/data2/pdb_mmcif/mmcif_files  \
 --obsolete_pdbs_path=/data2/pdb_mmcif/obsolete.dat \
 --pdb_seqres_database_path=/data2/pdb_seqres/pdb_seqres.txt \
 --uniprot_database_path=/data2/uniprot/uniprot.fasta \
 --num_multimer_predictions_per_model=$MULTIMER_PREDICTIONS_PER_MODEL \
 --model_preset=$MODEL_PRESET \
 --max_template_date=$MAX_TEMPLATE_DATE \
 --run_relax=True \
 --use_gpu_relax=True \
 --db_preset=$DB_PRESET > stdout.$PBS_JOBID 2> stderr.$PBS_JOBID
```

Sample Job Script (Multimer Reduced DBs)

For Alphafold2 **Multimer Reduced DBs** Batch Jobs

Note:

1. **Do not copy and paste** the job script in the next slide directly into your text editor.
2. Please **type it out manually** to avoid hidden characters.
3. Ensure that the jobscript **conforms to BASH syntax.**

```
#!/bin/bash

#PBS -P alphafold_project_name
#PBS -j oe
#PBS -N alphafold_job_name
#PBS -q volta_gpu
#PBS -l select=1:ncpus=10:mem=100gb:ngpus=1
#PBS -l walltime=20:00:00

cd $PBS_O_WORKDIR;
np=$(cat ${PBS_NODEFILE} | wc -l);

##------ THE ONLY PART FOR YOU TO CHANGE ------
## User settings
INPUT_FASTA_FILE_PATH=P10_trimer.fasta;   ## "my_abc123.fasta" is your input *.fasta file.
OUTPUT_DIR=`pwd`/alphafold22_output_1_m;   ## "alphafold_output_1" defines output folder name.

MAX_TEMPLATE_DATE='2022-03-30' # yyyy-mm-dd format
MULTIMER_PREDICTIONS_PER_MODEL=5
DB_PRESET=full_dbs     # db_presets: full_dbs, reduced_dbs
MODEL_PRESET='multimer' # model_preset: monomer, monomer_casp14, monomer_ptm, multimer

# Create output directory
mkdir -p ${OUTPUT_DIR}

##------  END TO CHANGE ------------------------

## Might not need to change until there is a newer version ##
## Change this to a newer alphafold container when it is released
IMAGE=/app1/common/singularity-img/3.0.0/alphafold/alphafold_v2.2.sif
## END ##

## DO NOT CHANGE ##
# Params and DB paths
ALPHAFOLD_DATA_PATH=/scratch2/biodata/alphafold/database/
ALPHAFOLD_DATA_PATH2=/scratch2/biodata/alphafold/database_v2_2/
ALPHAFOLD_MODELS=/scratch2/biodata/alphafold/database_v2_2/params

## Do not remove ##
mkdir -p alphafold/common
cp -n /scratch2/biodata/alphafold/setup_files/stereo_chemical_props.txt alphafold/common/
```

Orange is user configurable
Green is updatable
Black is fixed

Path to your fasta file.
No spaces allowed

Path to your desired output folder.
No spaces allowed.

Continued on next slide

Orange is user configurable
Green is updatable
Black is fixed

```
##--- To start & run Alphafold2 in Singularity container. ----
singularity run --nv \
 -B $ALPHAFOLD_DATA_PATH:/data \
 -B $ALPHAFOLD_DATA_PATH2:/data2 \
 -B $ALPHAFOLD_MODELS \
 -B .:/etc \
 --pwd `pwd` $IMAGE  \
 --data_dir=/data \
 --output_dir=$OUTPUT_DIR  \
 --fasta_paths=$INPUT_FASTA_FILE_PATH  \
 --uniref90_database_path=/data/uniref90/uniref90.fasta  \
 --mgnify_database_path=/data/mgnify/mgy_clusters.fa   \
 --small_bfd_database_path=/data2/small_bfd/bfd-first_non_consensus_sequences.fasta \
 --template_mmcif_dir=/data2/pdb_mmcif/mmcif_files  \
 --obsolete_pdbs_path=/data2/pdb_mmcif/obsolete.dat \
 --pdb_seqres_database_path=/data2/pdb_seqres/pdb_seqres.txt \
 --uniprot_database_path=/data2/uniprot/uniprot.fasta \
 --num_multimer_predictions_per_model=$MULTIMER_PREDICTIONS_PER_MODEL \
 --model_preset=$MODEL_PRESET \
 --max_template_date=$MAX_TEMPLATE_DATE \
 --run_relax=True \
 --use_gpu_relax=True \
 --db_preset=$DB_PRESET > stdout.$PBS_JOBID 2> stderr.$PBS_JOBID
```

Differing Flags

Additional DB paths/flags to include if using multimer
--pdb_seqres_database_path=/data2/pdb_seqres/pdb_seqres.txt \
--uniprot_database_path=/data2/uniprot/uniprot.fasta \

Additional DB path/flags to include is using monomer/not multimer
--pdb70_database_path=/data/pdb70/pdb70 \

Reduced DBs:
--small_bfd_database_path=/data2/small_bfd/bfd-first_non_consensus_sequences.fasta \

Configure MAX_TEMPLATE_DATE in 'yyyy-mm-dd' format to set the cutoff time point prior to the release date of structures.

Wrong:

```
image = /path/to/container/
INPUT_FASTA_FILE_PATH = abc.fasta
INPUT_FASTA_FILE_PATH = my abc.fasta
```

Correct

```
image=/path/to/container
INPUT_FASTA_FILE_PATH=abc.fasta
INPUT_FASTA_FILE_PATH=my_abc.fasta
```

PBS Job Scheduler

Submitting a Job

Steps

You have to run:

1. Prepare your fasta in your working directory
2. Create a PBS job script and save it in your working directory
    a. Example job scripts are in the following 2 slides
3. Submit PBS job script to PBS Job Scheduler

Server will run:

1. Job is in PBS Job Scheduler queue
2. Job Scheduler waits for server resources to be available
3. If available, Job Scheduler runs your script on remote gpu server

Submitting a Job

Save your job script (previous slides for examples) in a text file (e.g. train.pbs) then run the following commands

```
shell$ qsub train.pbs
675674.venus01
```

Job Status

```
shell$ qstat -xfn

venus01:
                                                                  Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
669468.venus01  ccekwk   volta    cifar_noco    --    1   1   20gb 24:00 F   --
    --
674404.venus01  ccekwk   volta    cifar_noco    --    1   1   20gb 24:00 F   --
   TestVM/0
675674.venus01  ccekwk   volta    cifar_noco    --    1   1   20gb 24:00 Q   --
    --
```

Statuses: Q(ueue), F(inish), R(unning), E(nding), H(old)

Submitting a Job

```
[ccekwk@atlas8-c01 classification]$ qsub train.pbs
697978.venus01
[ccekwk@atlas8-c01 classification]$ qstat -xfn

venus01:
                                                                  Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
695126.venus01  ccekwk   azgpu    cifar_noco    --    1   4   40gb 24:00 F   --
    --
697978.venus01  ccekwk   azgpu    cifar_noco    --    1   4   40gb 24:00 R   --
   TestVM/0*4
[ccekwk@atlas8-c01 classification]$
```

Statuses: Q(ueue), F(inish), R(unning), E(nding), H(old)

Job Chaining and Dependencies

Execute jobs in sequence

- **`qsub -W depend=afterok:<Job-ID> <JOB SCRIPT>`**
  - `qsub -W depend=afterany:836578.venus01 volta_benchmark.pbs`
- Job script <QSUB SCRIPT> will be submitted after the Job, <Job-ID> is successfully completed. Useful options to "depend=..." are:
  - **afterok**:<Job-ID> Job is scheduled if the Job <Job-ID> exits without errors or is successfully completed.
  - **afternotok**:<Job-ID> Job is scheduled if the Job <Job-ID> exited with errors.
  - **afterany**:<Job-ID> Job is scheduled if the Job <Job-ID> exits with or without errors

```
ccekwk@atlas9-c01 /hpctmp/ccekwk/tf_benchmarks/container$ qsub volta_benchmark.pbs
836578.venus01
ccekwk@atlas9-c01 /hpctmp/ccekwk/tf_benchmarks/container$ vim volta_benchmark.pbs
ccekwk@atlas9-c01 /hpctmp/ccekwk/tf_benchmarks/container$ qsub -W depend=afterany:836578.venus01 volta_benchmark
.pbs
836582.venus01
ccekwk@atlas9-c01 /hpctmp/ccekwk/tf_benchmarks/container$ qstat -fns1

venus01:
                                                            Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
836578.venus01  ccekwk   volta_lo gpu_tf_ben 392016   1  20   80gb 04:00 R 00:01 volta01/0*20
   Job run at Thu Apr 18 at 09:00 on (volta01:ncpus=20:mem=83886080kb:ngpu...
836582.venus01  ccekwk   volta_lo gpu_tf_ben     --    1  20   80gb 04:00 H   --    --
```

Job Chaining and Dependencies

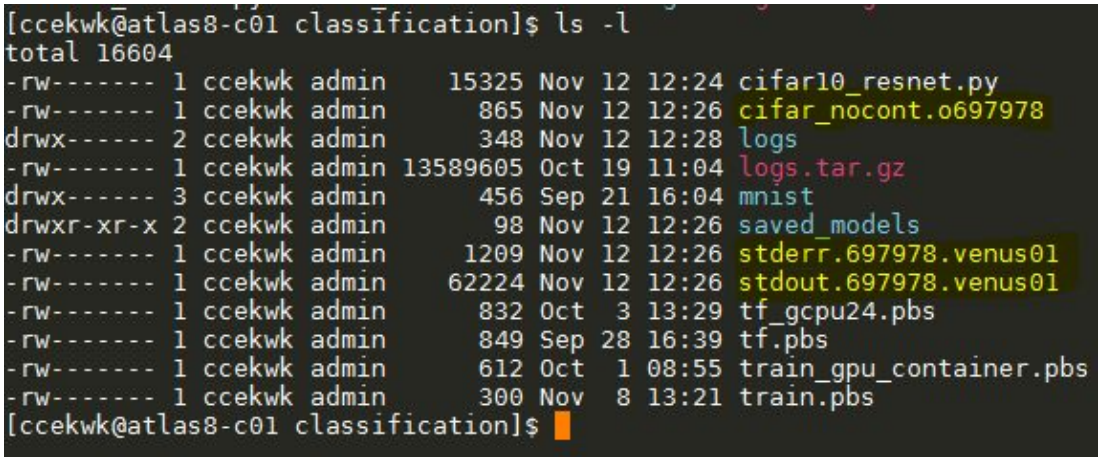
Useful PBS Commands

Action	Command
Job submission	`qsub my_job_script.txt`
Job deletion	`qdel my_job_id`
Job listing (Simple)	`qstat`
Job listing (Detailed)	`qstat -ans1`
Queue listing	`qstat -q`
Completed Job listing	`qstat -H`
Completed and Current Job listing	`qstat -x`
Full info of a job	`qstat -f job_id`

Checking Jobs

Log Files

- Output (stdout)
  - `stdout.$PBS_JOBID`
- Error (stderr)
  - `stderr.$PBS_JOBID`
- Job Summary
  - `job_name.o$PBS_JOBID`



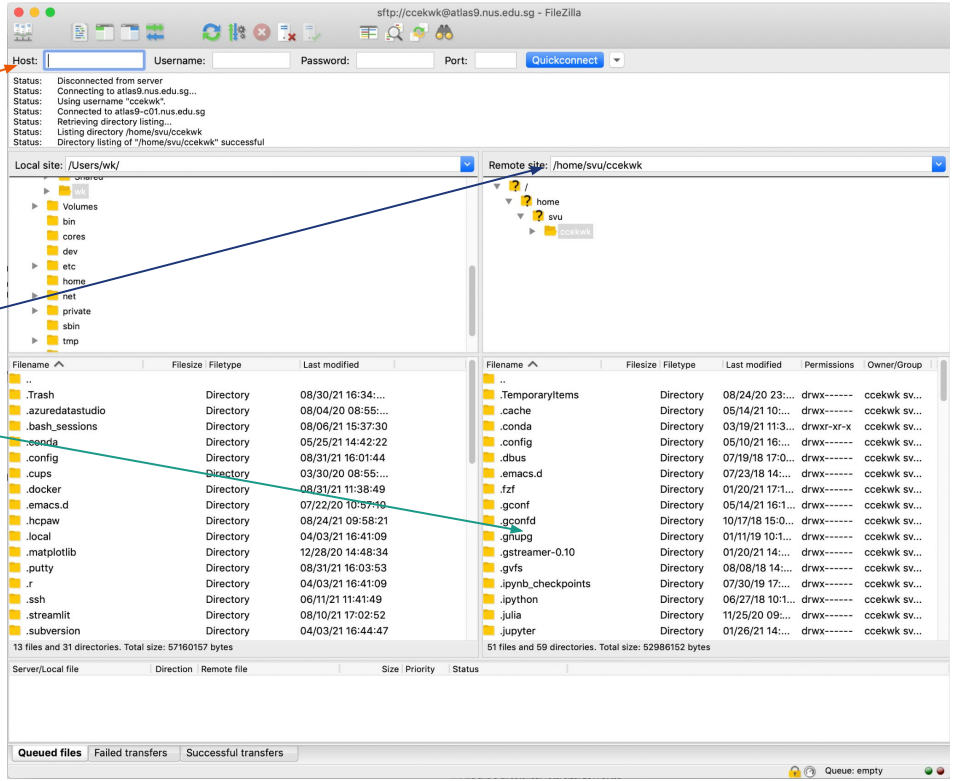
From Start to End

Setting up

```
ssh nusnet_id@atlas9

mkdir /scratch2/`whoami`

cd /scratch2/`whoami`

mkdir alphafold_workdir

cd alphafold_wokrdir

nano jobscript.txt  # opens a text editor

# Paste in sample job script available in /app1/common/alphafold/samples_jobscript

# Crtl+x -> y -> Enter | Save your jobscript
```

Uploading FASTA File(s)

1. Login (see slide 9)
2. Browse to `/scratch2/your_nusnet_id/alphafold_workdir`
3. Drag & Drop your .fasta file



Submitting your Alphafold2 Job

```
# Back to the terminal

qsub jobscript.txt

# Your job is now submitted

# Check job status

qstat -xfn
```

Job Complete, Retrieve Results

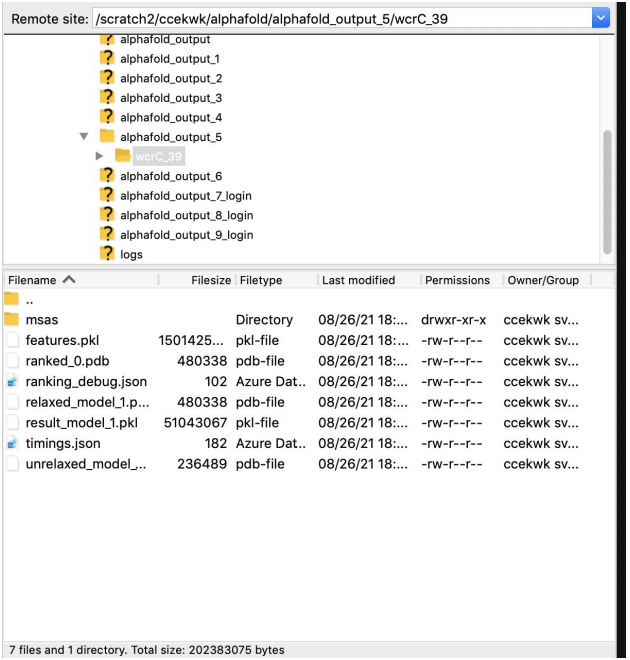
Remember your output directory set in the job script?

```
OUTPUT_DIR=`pwd`/alphafold_output_5
```

You can find it in:
`/scratch2/your_nusnet_id/alphafold_workdir/alphafold_output_5`

You can now download the output folder using filezilla

**OUTPUT_DIR might differ, please refer to the actual job script used*



Acknowledgement of Usage of NUS HPC Resources

Our primary mission is to provide the best of class, high-performance computing resources to support your computational research needs free of charge. To continuously improve the service, anticipate future demands, and keep track of our HPC facility's impact on the NUS research community in general, we request you to cite the REC team in your published research works.

Below is an example of a citation that may work for you:

"We would like to acknowledge that computational work involved in this research work is partially / fully supported by NUS IT's Research Computing group"

We would appreciate if you could send us a copy of your publication as well.

General Support

nTouch

https://ntouch.nus.edu.sg/ux/myitapp/#/catalog/home

Project/Research Collaboration or Long Term Engagement

Email

dataengineering@nus.edu.sg