# SCSE GPU CLUSTER TC1 USER GUIDE

Content	Page
Introduction	2
Logging into the cluster	4
Conda Package and Environment	8
SLURM User Guide	12
Guideline for SLURM Job Submission	15
Notice for Specific Coursework (CX4042)	20
Miscellaneous	21
Other Resources for Application	24
Important Notice	25

# Introduction



In SCSE GPU cluster TC1, there are two diverse types of nodes for diverse types of tasks. The initial node you log in to is called the **Head Node**, serving as the main access point for the cluster.

#### The GPU cards for computation are in those Compute Nodes, NOT in the Head Node.

The users are <u>not allowed</u> to compile and run code on the Head Node. The users must create the job script to submit the computation request (knowns as **non-interactive job**) to SLURM (**Simple Linux Utility for Resource Management**) job scheduler, for the system to process in the allocated GPU Compute Nodes.

The available resources for the user to utilise for GPU computation is limited by the assigned **QoS** (SLURM-Quality of Service).

Name of Program	Description & Purpose	Where to download
PuTTY	[SSH Client for Windows]PuTTY is an open-source software asSSH and Telnet Client for WindowsPlatform.SSH (Secure Socket Shell) is a networkprotocol that gives users, particularlysystem administrators, a secure way toaccess a computer over an unsecurednetwork.	The installer is available online and can be easily located using any search engine. Tip: Download the portable edition "putty.exe" - no need to install, ready for use http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html Other Online References: https://www.ssh.com/academy/ssh/putty
	[ How to SSH on MAC ] Online References: https://winonmacs.com/ https://www.servermania.com/kb/articles/ss	:h-mac/

#### **Client Tools for accessing the SCSE GPU Cluster Head Node**

-	· · ·	
WinSCP	[SFTP Client for Windows]	The official download site for WinSCP:
	WinSCP (Windows Secure Copy) is a	https://winscp.net/eng/index.php
	free and open-source SFTP, FTP, WebDAV and SCP client for Microsoft Windows	Installation tip: Select " <i>Explorer</i> " for user interface style
	SFTP (SSH File Transfer Protocol) is an	Other Online References:
	encrypted or secure file transfer	https://winscp.net/eng/docs/guide_install
	protocol.	https://www.puttygen.com/winscp
<b>File7ille</b>	[CETD Client for MAC]	Devenloed Cite for File7ille
Fliezilla	[SFIP Client for MAC]	Download Site for Fliezilla:
		https://filezilla-project.org/index.php
		Other online References of using SFTP on MAC:
		https://lemp.io/how-to-connect-to-sftp-mac-os/
		https://beebom.com/how-to-use-mac-terminal-ftp-sftp-client/
		https://tipsmake.com/use-terminal-on-mac-as-ftp-or-sftp-client

### Workflow for the user:

- 1. Establish your first access via SSH. Your home directory will only be created after your first login
- 2. Uploading all the necessary job script, coding, dataset from your host machine to the cluster's storage (your home directory) via SFTP client
- 3. Setup and load your Conda environment(s) with required applications for computation
- 4. Submitting the non-interactive job using SLURM scheduler
- 5. Copy the job output back to your host machine, via SFTP client

This guide assumes that you have basic knowledge of accessing the Linux system via command-line interface:

- Connecting to a Linux server via SSH
- Moving files to a Linux server from your local computer, and vice versa
- > The Linux basic commands such as: ls, cd, pwd, cp, mv, rm, chmod etc.

#### **References for Basic Linux Commands:**

https://serverdale.com/en/linux-commands https://centoshelp.org/resources/commands/linux-system-commands/ https://www.guru99.com/must-know-linux-commands.html

The GPU cards for your computation are NOT in the Head Node. So, do not attempt to execute command to acquire the GPU information in the Head Node e.g. "*nvidia-smi*", "*lspci* | grep -E 'VGA|3D'" and etc.

# Logging into the cluster

#### The SCSE GPU cluster is only accessible within NTU network

For off-campus access, the user must login to NTU Virtual Private Network (VPN) [ https://ntuvpn.ntu.edu.sg ]

Hostname of SCSE GPU Cluster Head Node	scsegpu-tc1
IP Address	172.21.148.240
Login Credential	Your NTU Network Account ID (in Lower Case) & Password

### SSH via PuTTY

1. Launch PuTTY.

Under the Host Name, ent	er the IP Address [ 172.21.148.240 ]	
Ensure that Port is 22.		
🕵 PuTTY Configuration		×
Category:		
	Basic options for your PuTTY sess	ion
Logging     Logging	Specify the destination you want to connect Host Name (or IP address)	to Port 22 O Serial Load Save Delete
	Olose window on exit: Always Never Only on clear	an exit
About	Open	Cancel

2. Examples of command-line access via your LINUX or MAC Terminal:

>> ssh -l <Your Username in Lower Case> 172.21.148.240
>> ssh -p 22 <Your Username in Lower Case>@172.21.148.240

On your first login, you will be asked to accept the host key.
 Do click "yes" to continue, and you should get a terminal window, where you will be prompted for your credentials. Log in with your NTU Network Account ID (*in Lower Case*) and password.



Best practice to exist the SSH session: type "exit" or press the keys Ctrl + D

#### SFTP via WinSCP/FileZilla – For file transfer between your local host machine and GPU cluster

- 1. In this example, the client tool using for file transfer is WinSCP.
- 2. Launch WinSCP. Check that the File Protocol is set to SFTP.
- 3. Under the Host Name, key in the IP Address [172.21.148.240]. Ensure that Port is 22. Key in your NTU Network
  - Account ID (in lower case) and password for the User name and Password field.



4. Ensure that you are in the correct directory before transferring your files over.

You can verify the pathname of your home directory with following command in SSH session:

>> pwd	
[scetest6@SCSEGPU-TC1:~]#	pwd
/home/UG/scetest6	
[scetest6@SCSEGPU-TC1:~]#	

Select the file that you want transfer and press the key "F5"

The *Upload-Window* will pop out to transfer the file from your host machine to your home directory. The *Download-Window* will pop out to transfer file from your home directory to selected local disk in your host machine.

Local Mark Files Comman	s Session Options Remote Help				
🖶 💱 🍃 Synchronize 📘	P R   P Queue -   Transfer S	Settings Default -	<i>8</i> ·		
📮 scetest6@172.21.148.240 🔅	🗳 New Session				
🏪 C: C-W10LTSC 🔹 🚰 🔹	🔽 •   🗢 • 🔶 • 🔁 🔁 😪	📙 scetestő 🔹 🚰 🔹 🔽 🔹 🔤	🕨 - 🔶 - 🛛 🔂 🟠 i	2 🖸 Find Fi	iles 💁
🙀 Upload 🔹 📝 Edit 🔹 💥	🛃 🕞 Properties 🛛 👻 🕂 🖃 🕅	Download - 📝 Edit - 🗙 (	🔏 🕞 Properties 📑 Ne	w •   + -	V
C:\_Tools\WinSCP-5.19.2-Porta	ole\	/home/UG/scetest6/			
Name ^ • ]] license.bt	Size Type i Parent directory 2 37 KB Text Document 2	Name		Size	Changed 14/12/2022 14/12/2022
ireadme.bxt WinSCP.com WinSCP.exe WinSCP.ini	Upload file 'readme.txt' to remote	directory:	? ×		
	Transfer settings Transfer type: Binary	for mana)			
	Transfer settings ▼ □ ₽0 not show this dialog box again In Commander interface the keyboard sh	OK Cancel	Help		
	to remean a life panel, dick here to go to	preferences.			
¢	>	<			>
B of 26.0 MB in 0 of 5		0 B of 0 B in 0 of 1			17 hidde

#### **Maintain Data in Your Directory**

- 1. Your data and computation are bounded within your assigned home directory and share folder (if any).
- 2. All home directories are set with disk quota limitation. The student users may have disk quota of 100GB, depending on their usage assignment.
- 3. The user may encounter following issues when the quota usage reached more than 98%:
  - Unable to login
  - Unable to compute data
  - > Unable to save in new data, or having data lost
- 4. You should do regular housekeeping, ensure there is sufficient free space for your computation.
- 5. Command to verify the disk usage at command line: ncdu

Scetest6@SCSEGPU-TC	1:~	-	>
ncdu 1.17 ~ Use the	arrow keys to navigate, press ? for help		
/home/UG/scetes	st6		 
1.3 GiB [######	HEEFERE		
66.0 KiB [	] /jobscript		
4.0 KiB [	]/.local		
2.0 KiB [	] /.ssh		
1.5 KiB [	] /.mozilla		
1.5 KiB [	] /.cache		
1.0 KiB [	] .bashrc		
1.0 KiB [	] .viminfo		
1.0 KiB [	] /.nv		
1.0 KiB [	] /.config		
1.0 KiB [	] /.jupyter		
512.0 B [	] .bash_profile		
512.0 B [	] .emacs		
512.0 B [	] .bash_history		
512.0 B [	] .kshrc		
e 512.0 B [	]/.ipython		
512.0 B [	] .k5login		
512.0 B [	] .bash_logout		
Total disk usage:	1.3 GiB Apparent size: 940.7 MiB Items: 42871		

Press the key "q" to abort disk scanning or exit.

*Reference on ncdu:* https://ostechnix.com/check-disk-space-usage-linux-using-ncdu/

- 6. Best practices to ensure enough disk space for your computation:
  - Regularly check on the disk usage for your home directory
  - > Transfer and backup your data to your personal device
  - Remove unwanted data in your home directory
  - Do not remove those system directories and files, which naming with a "." in front, such as ".bash\_profile", ".bashrc", ".config" etc
  - > The purpose of the system folder ".conda" is to store the packages and environments that you have setup and installed for your computation. Remove those unwanted environments to free the disk space.

# Conda Package and Environment

The module package tool is available on the GPU Cluster, allowing users to easily configure their environment based on the application needed. As this cluster is shared among students of different courses in SCSE, there may be some applications that are not relevant to you.

To view the available share applications, apply the modules under /cm/shared/modulefiles

>> module avail			
[scetest6@SCSEGPU-TC1:~]	avail		
boost/1.71.0 cmd cmsh cluster-tools/9.0 cmjob freeipn	gcc/8.4.0 luajit j/l.6.4 gcc/9.2.0 module-ir	<pre>/modulefiles openldap python37 nfo python3 shared</pre>	
	/cm/shared	i/modulefiles	
anaconda	fftw3/openmpi/gcc/64/3.3.8	intel-tbb-oss/ia32/2020.3	python/3.7.12
blacs/openmpi/gcc/64/1.1patch03	gcc/8.5.0	intel-tbb-oss/intel64/2020.3	python/3.9.13
blas/gcc/64/3.8.0	gcc/10.4.0	iozone/3 487	python/3.10.5
cuda/10.2	gdb/8.3.1	lapack/gcc/64/3.8.0	R
cuda/11.7	go	matlab/R2020a	<pre>scalapack/openmpi/gcc/2.1.0</pre>
default-environment	hdf5 18/1.8.21	miniconda-py37	singularity
fftw2/openmpi/gcc/64/double/2.1.5	hp1/2.3	miniconda-py39	
fftw2/openmpi/gcc/64/float/2.1.5 [scetest6@SCSEGPU-TC1:~]#	hwloc/1.11.11	openmpi/gcc/64/1.10.7	

To view the description of a module:

### **Setup Your own Conda Environment**

In this GPU Cluster, the user has no right to execute Sudo and install application to the system folders.

You may download and install the Anaconda or Miniconda in your home directory for operation.

Reference links:

https://www.anaconda.com/products/distribution https://docs.conda.io/en/latest/miniconda.html

OR use the available Anaconda in this GPU Cluster:

To load the module of **anaconda**:

>> module load anaconda

To view the information of the loaded anaconda:

>> conda info

**The Conda commands can only run after the module is loaded.** By default, the read-only base environment is activated. There are ready packages in the centralised Anaconda (or the base environment), you may issue the following command to view the list:

>> conda list

You are **<u>not allowed</u>** to install any packages in the base environment. If you find those packages in the base environment do not meet your requirement, you may setup own conda environment in the home directory, and install the require packages for your operation.

For first-time running Conda, the system may prompt you to execute the command "conda init <shell>".

Jpdate [ 19 January 2023 ]	
----------------------------	--

The purpose of the command is to write some shell code into the system file (*e.g.* ~/.*bashrc*) as startup script for conda. For default shell using bash, execute the following

#### >> conda init bash

For the change to take effect, execute

>> source .bashrc

#### **Commands to manage the Conda environment:**

Description	Examples of Command Execution
To create a new Conda environment naming as "TestEnv"	>> conda create -n TestEnv
Activate the environment "TestEnv"	>> conda activate TestEnv
To exit the environment "TestEnv"	>> conda deactivate
List the environment created in your home directory	>> conda env list
To export the configuration of an environment into a <i>.yml</i> file.	>> conda env export > TestEnv.yml
To create the environment from the existing .yml file	>> conda env create -f TestEnv.yml
To update the content of an existing .yml file, with option "prune" to remove the outdated configuration and dependencies	>> conda env updateprefix ./TestEnvfile TestEnv.ymlprune
To remove the environment " <i>TestEnv</i> "	>> conda env remove -n TestEnv
Reference on managing the environment:	

https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html#creating-an-environment-with-commands

#### Commands to manage the variables in the environment:

Description	Examples of Command Execution
List the variables in the environment	>> conda env config vars list
To set a new variable in the environment "my_var1"	>> conda env config vars set my_var1=value
To unset (remove) the variable "my_var1"	>> conda env config vars unset my_var1
Always re-activate the environment after adding a new variable and removing a variable	>> conda activate TestEnv

10 | SCSE GPU CLUSTER TC1 (UG) USER GUIDE

[scetest6@SCSEGPU-TC1] # conda activate TestEn	v
(TestEnv) [scetest6@SCSEGPU-TC1]# conda env c	onfig vars list
(TestEnv) [scetest6@SCSEGPU-TC1]# conda env co	onfig vars set my_varl=10
To make your changes take effect please react.	ivate your environment
(TestEnv) [scetest6@SCSEGPU-TC1] # conda activ	ate TestEnv
(TestEnv) [scetest6@SCSEGPU-TC1]# conda env c	onfig vars list
my_varl = 10	
(TestEnv) [scetest6@SCSEGPU-TC1] # conda env co	onfig vars unset my_varl
To make your changes take effect please react.	ivate your environment
(TestEnv) [scetest6@SCSEGPU-TC1] # conda activ	ate TestEnv
(TestEnv) [scetest6@SCSEGPU-TC1] # conda env c	onfig vars list
(TestEnv) [scetest6@SCSEGPU-TC1] #	

### Commands to manage the packages in the environment

The user must always create a new environment to install the package for own use.

Description	Examples of Command Execution
Search for available packages. For this example,	>> conda search python
search for available python to install	>> conda search python=3.9
To search and install available package from	# To search for a package in 3 <sup>rd</sup> party channel "conda-forge"
third-party channel	<pre>&gt;&gt; conda search -c conda-forge <name of="" package=""></name></pre>
The conda package manager usually installs the package from the official default channels. You must specify the channel in the command if to search and install package from third-party channel <b>conda-forge</b> is one of the third-party channels, providing latest conda packages.	# To install the package from the selected third-party channel >> conda install -c conda-forge <name of="" package=""></name>
Reference links: https://docs.conda.io/projects/conda/en/latest/user- guide/concepts/channels.html https://ostechnix.com/enable-conda-forge-channel-for- conda-package-manager/	
List the installed packages	>> conda list
Install selected version of the package	>> conda install python=3.9.7
The GPU Compute Nodes are ready with their own version of Cuda by NVIDIA, which may not be up-to-date or applicable for user's operation. The user may setup own Conda Environment, and install the desired version of Cuda Toolkit for operation.	<pre>[scetest68SCSEGPU-TC1]# module avail   grep cuda anaconda</pre>
Reference links: https://docs.nvidia.com/deploy/cuda- compatibility/index.html https://anaconda.org/nvidia/cudatoolkit	<pre>&gt;&gt; conda search -c nvidia cudatoolkit # Then install the select version for operation &gt;&gt; conda install cudatoolkit=<version> &gt;&gt; conda install -c nvidia cuda-toolkit=<version></version></version></pre>
Remove a package	<pre>&gt;&gt; conda uninstall python=3.9</pre>

Update	[ 19 Januar	y 2023 ]
--------	-------------	----------

Reference links on package installation in conda:

https://docs.anaconda.com/anaconda/user-guide/tasks/install-packages/ https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-pkgs.html#installing-packages

#### **IMPORTANT NOTE:**

Advise to install the packages using *conda install*. Only use *pip install* if the required package is not available in conda. This is to prevent package incompatibility.

If you want your conda environment to be loaded each time you start your SSH session, you may append "module load anaconda" and "conda activate yourenvname" at the end of the system file "**.bashrc**", located at your home directory.



Remove unwanted Conda environment, to free disk space

- 1. Exit from the conda environment >> conda deactivate
- 2. List the conda environment in your home directory >> conda env list
- Only remove unwanted environment in your home directory>> conda env remove –name <Name of ENV to remove>

```
[scetest6@SCSEGPU-TC1] # conda env list
# conda environments:
-
base
                      * /apps/anaconda3
TC
                         /home/MSAI/scetest6/.conda/envs/TC
                         /home/MSAI/scetest6/.conda/envs/TestEnv
TestEnv
                         /home/MSAI/scetest6/.conda/envs/envtest
envtest
                         /home/MSAI/scetest6/.conda/envs/mm21ab
mm21ab
                         /home/MSAI/scetest6/.conda/envs/mmlab
mmlab
py38-tfgpu
                         /home/MSAI/scetest6/.conda/envs/py38-tfgpu
[scetest6@SCSEGPU-TC1] # conda env remove --name envtest
Remove all packages in environment /home/MSAI/scetest6/.conda/envs/envtest:
```

## **SLURM User Guide**

**SLURM** (*Simple Linux Utility for Resource Management*) is a software package for submitting, scheduling, and monitoring jobs on large computer clusters.

In this GPU cluster, you are **not allowed** to compile and run your code on the head node. You must create a job script to submit your computation request (**non-interactive jobs**) to SLURM Scheduler, for the system to process in allocated GPU Compute Nodes.

The resources for GPU computation is limited by the **QoS** (SLURM-Quality of Service) assigned to you.

Command to view the QoS assigned to the user >> sacctmgr show user <username> withassoc format=user,qos

# You may copy and paste the commands to the command prompt in your SSH session window for execution # Replace the text "<username>" with your own username

[scetest6@SCSEGPU-TC1]#	sacctmgr	show	user	scetest6	withassoc	format=user, qos
User	QOS					
scetest6	normal					
[scetest6@SCSEGPU-TC1]#						
			"			

# The above example shown the user "scetest6" can use the assigned QoS "normal" for GPU computation

Command to show the resources configured for the QoS:

>> sacctmgr -P show qos <Name of QoS> withassoc format=name,MaxTRESPU,MaxJobsPU,MaxWall

>> sacctmgr show qos <Name of QoS> withassoc format=name%+15,MaxTRESPU%+40,MaxJobsPU%+10,MaxWall%+10

# %+<value> is to add value to expand the viewing field



Example showing the Resources assigned to QoS name "normal":

- MaxTRESPU (Maximum Trackable RESources Per User): cpu (Maximum number of CPUs/CORES for the user to deploy) = 20 gres/gpu (Maximum number of GPU for the user to deploy) = 1 mem (Maximum size of Memory for the user to deploy) = 64G
- 2. MaxJobsPU (Maximum number of Jobs Per User, can run at a given time) = 2
- 3. MaxWall (Maximum Wall clock time per user, to run the job, DD-HH:MM:SS) = 6-hours

The SLURM job script is required, to specify the necessary resources, application, and path to execute your code. Then, submit the job script to SLURM at command line.

References links on creating the SLURM job script:

https://slurm.schedmd.com/quickstart.html

https://www.carc.usc.edu/user-information/user-guides/hpc-basics/slurm-templates

https://svante.mit.edu/use\_slurm.html

<pre>sccntrol # View the details of a numing job based on its jobid and showing the id of GPU card in the node running the job, under the line "Nodes=SCSEGPU" &gt; scntrol show -d jobid <jobid< th=""><th>Command</th><th>Definition (#) &amp; Lise Examples (&gt;&gt;)</th></jobid<></pre>	Command	Definition (#) & Lise Examples (>>)
<pre>interval job, under the line "Nodes=SCSEGPU" &gt;&gt; scontrol show-d jobid (jobid&gt;</pre>	scontrol	# View the details of a running job, based on its jobid and showing the id of GPU card in the node running the
<pre>&gt;&gt; scontrol show -d jobid <jobid> &gt;&gt; scontrol show -d jobid <jobid> * View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # view the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # view the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # view the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01:U] # view the status of the node sing the status of the node of the status of the node in the node in the status of the node in the status of the node in the status of the node in the node. # Display state of the compute nodes in respective partition &gt;&gt; sinfo # Display state of the compute nodes in respective partition &gt;&gt; sinfo # Display state of the compute nodes in the node inder NODELIST SCSEGPU_TC1 = [04:U] FarstMine to howing "the" status of the node inder NODELIST SCSEGPU_TC1 = [04:U] # To view worr own job history &gt;&gt; sacct -a -format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 # To view worr own job history &gt;&gt; sacct -a -format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 # Job viewing excluding those lines with "batch" &gt;&gt; sacct -a -format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+</jobid></jobid></jobid></jobid></jobid></jobid></pre>		inh under the line "Nodes=SCSEGPU"
<pre>Provide the set of the comparation of the set of t</pre>		>> scontrol show -d jobid <jobid></jobid>
Numbiodes: Numerica: Numerica: CFU/TASL: RegISTCT:C0:0:::*           Numbiodes: Numerica: Numeric		
<pre>index-equipation is in the provide in the provide interpretation in the interpretation in the interpretation in the provide interpretation in the provide interpretation in the provide interpretation in the provide interpretation interpretati</pre>		NumNodes=1 NumCPUs=1 NumTasks=1 CPUs/Task=1 RegB:S:C:T=0:0:*:*
Hodes=CSEGF-TG-101 CFT_IB=-0-1 KEN=1203 GRES=pup (IXX:0)           Hubble State           Hubble State           Febtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           Pebtures=(mill)           CFCN1[ord=CCUCade679           AvailableFeatures=(mill)           Great-pup (Si3o-1)           OrdeHame=SCEGFU-TCL-10           Very the status of the nodes in the Partition "SCEGFUF-TCL-10" Version=19.05.5           Operations: State=WIKTN           State=WIKTN           Pattice=T(mill)           Great-pup (Si3o-1)           NodeMatrice=T(mill)           Great-pup (Si3o-1)           State=WIKTN           Pattice=Treate the resource status of the node, based on the information at the lines "GpTRES" and "AllocTRES".           The values shown under "AllocTRES" are the resources bing allocated to the running jobs in the node           Vocare metademetation           State=WIKTN           Vocare metademetation           VariableSCSEGFUTCL]           State=Treate the resource status of the node, based on the information at the lines "		TRES=cpu=1,mem=12G,node=1,b111ing=1,gres/gpu=1 Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
Interview of the end of		Nodes=SCSEGPU-TC1-01 CPU_IDs=0-1 Mem=12288 GRES=gpu(IDX:0)
OverSubscribt-OK Consignous-0 Licenser(unil) Metwork=(unil)           # View the status of the nodes in the Partition "SCSEGPU_UG"           >> scontrol show node SCSEGPU-TC1-[01-07]           NodeName=GCSEGPU-TC1-10 Arch=x86_64 CoresPerSocket=16 (CPURAIIco-2C CPUTot=64 CPULoad=6.79 AvailableFeatures=(unil) Gress-pqu:8(3:o-1) RodeMatm=SCSEGPU-TC1-10 NodeHostName=SCSEGPU-TC1-10 Version=19.05.5 OS=Linux 3.10.0-D062.11.1.el7.x86_c44 [13:BNP Fri Sep 13:2155:14 Urc2 2019 Resiltency=Table Alicokem=3848 Orket=~2 Board cart2 Available Partime/SCSEGPU-TC1-10 NodeHostName=SCSEGPU-TC1-10 Version=19.05.5 OS=Linux 3.10.0-D062.11.1.el7.x86_c44 [13:BNP Fri Sep 13:2155:14 Urc2 2019 Resiltency=Table Alicokem=3848 Orket=~3 DestTime=0711ae>D27D30132 ShitmedKem=3848 Orket=~2 Board cart2 Available Partime>Urc2 ScSEGPU-TC1-12-227D30132 ShitmetArc11me=2020121-20013148:43 Cortex=WatersD-Xx94Kt==0 ExtSensorsJoules=n/s ExtSensorsPatts=0 ExtSensorsTemp=n/s You can reference the resource status of the node, based on the information at the lines "CfgTRES" and "AllocTRES". The values show under "AlloCTRES" are the resources being allocated to the running jobs in the node.           # JDsplay state of the compute nodes in respective partition > sinfo         # Display state of the compute nodes in respective partition > Sinfo           # To view your own job history > Sacct         # To view your own job history > sacct - format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 # To view used mon job history > sacct - a - format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 # Vob viewing.excluding those lines with "batch" >> sacct - a - format=user.jobid.jobname%+15.gos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 propww batch # Use the jobs executed by specific user <username> &gt;&gt; sacct - a - fo</username>		Features=(null) DelayBoot=00:00:00
<pre># View the status of the nodes in the Partition "SCSEGPU_UG" &gt;&gt; scontrol show node SCSEGPU-TC1-[01-07] NodeMame=SCIEGPU-TC1-10 Arch+x86 64 CoresPerSockt=16 (CVRLlco=2c CPTCo=64 CURLoad=6.79 AvailableFeatures=(null) ActiveFeatures=(null) Gree=mpu:8(3:0-1) Gree=mpu:8(3:0-1) Gree=mpu:8(3:0-1) Gree=mpu:8(3:0-1) State=VIKED ThreadBretCore=2 TmpDisFro Weight=1 Owner=W/A MCS_label=W/A Partition=SCSEGPU-TC1-10 NodeMostName=SCSEGPU-TC1-10 Version=19.05.5 OS=Linux 3.10.0-1062.11.e17.x86_64 #1 SMP Fri Sep 13 22:55:44 UTC 2019 RealMemory=18508 Allockme=S408 MostLame=2CSEGPU-TC1-10 Version=19.05.5 OS=Linux 3.10.0-1062.11.e17.x86_64 #1 SMP Fri Sep 13 22:55:44 UTC 2019 RealMemory=18508 Allockme=S408 MostLame=2CSEGPU-TC1-10 Version=19.05.5 OCTILE2=opu=26.memeS4800M, great/gpu=2 Cartex=WIKED ThreadBretCore=2 TmpDisFro Meight=1 Owner=W/A MCS_label=W/A BootTime=2021-12-22T0913123 BlundBitedTime=2021-12-20T13148443 Cartex=Niket=0 Awatats=0 Cartex=tofEx=opu=26.memeS4800M, great/gpu=2 Cartex=Niket=0 Cartex=tofESCSEGPU=TC1] Cartex=tofESCSEGPU=TC1] Cartex=tofESCSEGPU=TC1] ScsEGPU_USALE=== Tripinter = 1 file ScSEGPU=TC1-[01-03] ScSEGPU_USALE up infinite = 1 file ScSEGPU=TC1-[01-03] ScSEGPU_USALE up infinite = 1 file ScSEGPU=TC1-[04-07] ScSEGPU_USALE up infinite = 1 file ScSE</pre>		OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
<pre>&gt;&gt; scottrol show node SCSEGPU-TC1-[01-07] NodeName=SCSEGVU-TC1-10 Archwsk6 64 CoresPerSocket=16 CFW11co-36 CFW0-ref CFW0-ref CFW0-ref CFW0-ref-6.79 ArailableTeaturese (null) Greamppus (3:0-0-1062.1.1.suf7.x86 64 f1 SWP Fri Sep 13 22:55:14 UTC 2019 RealMemory12508 Allockeme5480 Freekeme348 Sockerse7 Benatds=1 State=MIXED ThreadsPerSore=7 TmpDisk=0 Weight=1 Owner=W/A MCS_label=W/A Partitions=CSEGVU-MSh1 BootTime=901-12-2770913122 SlumdStarTImm=2021-12-0T13:48:43 CfgTRES*opu=42, meme53480M, gres/gpu=7 CapMatter=/A CurrentMatter0 AreWAtter=0 ExtSensorsJoulesm/A ExtSensorsMatte=0 ExtSensorsTemp=n/s You can reference the resource status of the node, based on the information at the lines "CfgTRES" and "AllocTRES". The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node. # Display state of the compute nodes in respective partition &gt;&gt; sinfo ScSEGFU_UG up infinite 3 mix SCSEGFU-TC1-[01-03] SCSEGFU_UG up infinite 4 idle SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 4 idle SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 4 mix SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 5 mix SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 4 mix SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 4 mix SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 4 mix SCSEGFU-TC1-[04-07] SCSEGFU_UG up infinite 7 &gt; sacct -format=user.jobid.jobname%+15.qos.alloctres%+50.nodelist.start.elapsed.state.reason%+20 # To view wal job history for the day.option "-a" &gt;&gt; sacct -cormat=user.jobid.jobname%+15.qos.alloctres%+50.nodelist.start.elapsed.state.reason%+2</pre>		# View the status of the nodes in the Partition "SCSEGPU_UG"
NodeName=SCSEGPU-TC1-10 Arch=x86_64 CoresPerSocket=16         CFUENLoc=26 CFUTor=46 CFUTora=67         Wailable?Eduttes=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         Active?eatures=(null)         RealMemory=128008 Allockem<8300 FreeKem>304 Sockets=2 Socrad=1         State=MXKD_IntreadsPerCore=2 TmpDist=V Mexpht=1 Conte=TVA KCS_labe1=N/A         Bootline=7041-12-27003130 SlumdEtatTime=2021-12-30T13:48:43         Ccfpr#ds=cpu=26, mem=1245004, grees/qpu=2         CurrenMatts=0 AveNatts=0         Vacan reference the resource storue of the node, based on the information or the lines "CfpTRES" and "AllocTRES". The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         # Display state of the compute nodes in respective partition         > sinfo         [sccEseFU_UG       up       infinite         ScSEGFU_UG       up       infinite         For SIATE not showing "Ide" or "mk", may indicate the node under NODELIST         ScSEGFU_UG       up       infinite         For SIATE not showing "Ide" or "mk", may indicate the node under NODELIST has been down      <		>> scontrol show node SCSEGPU-TC1-[01-07]
CEUALIC-26 CEPTOL-64 CEPLOAD=6.79         AvailableFeatures=(null)         ActiveFeatures=(null)         Corsergue (3:50-1)         NodeAdd=SCSEGEU-TC1-10 NodeKostName=SCEGEU-TC1-10 Version=19.05.5         SoS=Linux 3:10.0-1062(1).1e17.x66(4 \$1 SME Fri Sep 13.22155:44 UTC 2019         RealMemory=128508 AllocKemeS380 FreeKeme3380 Sockts=2 Reades1         StateMEND ThreadSerCore2 TmpDish=0 Weight=1 Owner=N/A MCS_Label=N/A         Partition=SCSEGUE MSAI         BooTime=201-12-270709131228 SlumedSartime=2021-12-30713148:43         CfGTEES=ropu=6, mem=12850M, billing=6, gres/gpu=7         CopMatts=ru/a         ChrestMatts=0 AveNatts=0         ExtSensorsJoules=ru/a         ExtSensorsJoules=ru/a         Sinfo         // Display state of the compute nodes in respective partition         >> sinfo         [sccEegU_MSAI         [sccEegU_MSAI         PATITION         Notaresru/a         ScSEGEU_UG         Up       infinite         ActiveScEEGEU_MSAI         PartitioN         ScSEGEU_MSAI         ScSEGEU_UG       up         Infinite       mix SCSEGEU-TC1-[01-03]         ScSEGEGU_MSAI       up         ScSEGEU_MSAI       mix         ScSEGEU_MAI       u		NodeName=SCSEGPU-TC1-10 Arch=x86 64 CoresPerSocket=16
AvailableFeatures=(null) ActiveFeatures=(null) Gress=gpui8(510-1) NodeAddrSCSGRU-TC1-10 NodeHostName=SCSEGFU-TC1-10 Version=19.05.5 OS-Linux 3.10.0-1062.1.1.e17.x86.64 #1 SNF Fri Sep 13 22155144 UTC 2019 RealMenory:12508 Allock=me63496 Sockets=2 Boards=1 State=VIKUD ThreadsPerCores2 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A Partition=SCSGRU NSA1 BooTime=2021-12-2709:13:23 SlumdStartTime=2021-12-30T13:48:43 CfgTESS=cpu=4; mem-13505M, billing=64, gress/gpu=3 AllocTRES=cpu=26, mem=63480M, gress/gpu=7 CapMatts=rm/a CurrentWatts=0 AveNatts=0 ExtSensorsJoules=r/s ExtSensorsWatts=0 ExtSensorsTemp=n/s         You can reference the resource status of the node, based on the information at the lines "CfgTES" and "AllocTRES". The values shown under "AlloCTRES" are the resources being allocated to the running jobs in the node.         # Display state of the compute nodes in respective partition > sinfo       Sinfo         ScSEGEU_0G       up infinite       3 mix SCSEGEU-TC1-[01-03] SCSEGEU_0G         ScSEGEU_0G       up infinite       4 mix SCSEGEU-TC1-[04-07] SCSEGEU_NSAT         ScSEGEU_NSAT       up infinite       4 mix SCSEGEU_TC1-[04-07] SCSEGEU_NSAT         ScSEGEU_NSAT       up infinite       4 mix SCSEGEU_TC1-[04-07] SCSEGEU_NSAT         ScsEGEU_NSAT       up infinite       4 mix SCSEGEU_TC1-[04-07] SCSEGEU_NSAT         ScsEGEU_NSAT       up infinite       4 mix SCSEGEU_TC1-[06-11]         For STATE not showing "Idle" or "may indicate the node under NODELIST has been down       > sact         * Job viewing, excluding those lines with "batch"       > sact         * Job		CPUAlloc=26 CPUTot=64 CPULoad=6.79
Accessed (min)         Greessed()         Greessed()         NodeAddr=SC2EGPU-TC1-10         NodeAddr=SC2EGPU-TC1-11         Scate=StateScateScateScateScateScateScateScate		AvailableFeatures=(null)
NodeAdd=SCSEGPT-TC1-10 NodeHostNameSCSEGPT-TC1-10 Version=100.5         OS-Linux 3.10.0-1062(1.1):e17.XeG 64 f1 SIXPFT1 Sep 122:55:44 UTC 2019         RealMemory=125508 AllocMameG3800 Encedem3384 Sockets=2 Boards=1         State=M1XED ThreadeBerCore2 TmpDisk=0 Weight=1 Owner=W/A MCS_label=N/A         Bootline=2011-12-220011333 SlumdStateTime=2021-12-30T13:48:43         CfgTRES=cpu=64, mm=125508, blling=64, grea/gpu=7         AllooTRES=cpu=64, mm=125508, blling=64, grea/gpu=7         CapWatt=#/A         CurrentWatts=0         ExtSensorsJoule=m/A XeWatts=0         ScSEGPU_TC1 # Sinfo         PARTITION       AVAIL TIMELIMIT NODES STATE NODELIST         ScSEGPU_UG       up infinite 4 inite SCSEGPU-TC1-[04-07]         ScSEGPU_MSAI       up infinite 4       mix SCSEGPU-TC1-[04-07]         ScSEGPU_MSAI       up infinite 4       mix SCSEGPU-TC1-[04-07]         ScSEGPU_MSAI       up infinite 4       mix SCSEGPU-TC1-[04-07]         ScSEGPU_MSAI <td></td> <td>Accivereatures=(hull) Gres=gpu:8(S:0-1)</td>		Accivereatures=(hull) Gres=gpu:8(S:0-1)
OS=Linux 3.10.0-1062.1.1.e17.x86 64 41 SMP Fr1 Sep 13 22155:44 UTC 2019         RealMemory=125508 Allockeme/3040 Freekees=2 Boards=1         State=MIXED ThreadsPerCore=2 TmpD1sk=0 Weight=1 Owner=N/A MCS_label=N/A         Partition=SCSEGU MSh1         BootTime=2021-12-2709113:23 SlurmdStartTime=2021-12-30T13:48:43         CfgTERS=ropu=64, mem:128508M, billing=64, gres/gpu=7         CapRateS=V/A         CurrentWatts=0 AveRatts=0         ExtSensorsJoules=m/s ExtSensorsWatts=0 ExtSensorsWeight=0         ExtSensorsJoules=m/s ExtSensorsWatts=0         SensorsJoules=m/s ExtSensorsWatts=0         ExtSensorsJoules=m/s ExtSensorsWatts=0         SensorsJoules=m/s ExtSensorsWatts=0         SensorsJoules=m/s ExtSensorsWatts=0		NodeAddr=SCSEGPU-TC1-10 NodeHostName=SCSEGPU-TC1-10 Version=19.05.5
Ketaledady=12303 Allocade=0300 Predede=0333 Societe5* Doited5*		OS=Linux 3.10.0-1062.1.1.el7.x86_64 #1 SMP Fri Sep 13 22:55:44 UTC 2019
Partitions=SCSECPU MSAT         BootTime=2021=12-22709:13:23 SluredStartTime=2021=12-30T13:48:43         CrgTRES=cpu=64, mem=128506M, billing=64, gres/gpu=7         CapMatts=r/a         CurrentWatts=r/a         Sinfo         # Display state of the compute nodes in respective partition         >> sinfo         [scetest6@SCSEGPU-TC1] # sinfo         PARTITION       AVALL TIMELINIT NODES         SCSEGPU_UG       up       infinite         3 mix SCSEGPU_TC1=[01-03]       SCSEGPU_TC1=[04-07]         SCSEGPU_MSAT       up       infinite         4 mix scSEGPU_TC1=[04-07]       SCSEGPU_TC1=[04-07]         SCSEGPU_MSAT       up       infinite         4 mix scSEGPU_TC1=[04-07]       SCSEGPU_MSAT         ScSEGPU_UG       up       infinite         4 mix scSEGPU_TC1=[04-07]       SCSEGPU_MSAT         ScSEGPU_UG       up       infinite		RealMemory=128508 AllocMem=63480 FreeMem=3384 Sockets=2 Boards=1 State=MIXED ThreadsPerCore=2 TmpDisk=0 Weight=1 Owner=N/A MCS label=N/A
BootTime=2021-12-27T09:13:23 SlurmdStartTime=2021-12-30T13:48:43         CfgTRES=opu=64, mem=63480M, gres/gpu=7         CapMatts=n/a         CurrentNatts=0 AveNatts=0         ExtSensorsMatts=0 ExtSensorsWatts=0         ExtSensorsMatts=0 aveNatts=0         Free values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         # Display state of the compute nodes in respective partition         >> sinfo         # Display state of the compute nodes in respective partition         >> sinfo         [scsEGPU_UG       up         Infinite       1 mix SCSEGPU-TC1-[01-03]         SCSEGPU_UG       up         Infinite       1 mix SCSEGPU-TC1-[04-07]         SCSEGPU_SAIL       up         If or view your own job history         >> sacct       # To view your own job history         >> sacct       # To view your own job history for the day, option ".a"         >> sacct       -format=user.jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"       >> sacct -aformat=user.jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"       >> sacct -aformat=user.jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job		Partitions=SCSEGPU_MSAI
Ctgttc5=cpu=c4, mem=258008, pilling+c4, gte8/qpu=3         AllocTES=cpu=c2, mem=c38008, gte8/qpu=3         CurrentWatts=0 AveWatts=0         ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s         You can reference the resource status of the node, based on the information at the lines "CfgTRES" and "AllocTRES". The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         # Display state of the compute nodes in respective partition         >> sinfo         # ScSEGPU_UG       up infinite         generation         SCSEGPU_UG       up infinite         generation         SCSEGPU_UG       up infinite         for state not showing "def" or "mix", may indicate the node under NODELIST SCSEGPU_NSAI       scSEGPU-TC1-[01-03]         SCSEGPU_UG       up infinite       mix SCSEGPU-TC1-[04-07]         SCSEGPU_UG       up infinite       mix SCSEGPU-TC1-[04-07]         SCSEGPU_MSAI       up infinite       mix SCSEGPU-TC1-[04-07]         SCSEGPU_MSAI       up infinite       mix SCSEGPU-TC1-[04-07]         SCSEGPU_UG       up infinite       mix SCSEGPU-TC1-[04-07]         SCSEGPU_MSAI       up infinite       mix SCSEGPU-TC1-[04-07]         ScSEGPU_TC1 not howing "def" or "mix", may indicate the node under NODELIST has been down       scate - format=user, jobid, jobname*+15, qos, alloctres*+50, nodelist, s		BootTime=2021-12-22T09:13:23 SlurmdStartTime=2021-12-30T13:48:43
CapWatts=n/a CurrentWatts=0 AveWatts=0 ExtSensorsJoules=n/s ExtSensorsTemp=n/s You can reference the resource status of the node, based on the information at the lines "CfgTRES" and "AlloCTRES". The values shown under "AlloCTRES" are the resources being allocated to the running jobs in the node. # Display state of the compute nodes in respective partition >> sinfo # Display state of the compute nodes in respective partition >> sinfo [scetest68SCSEGEU-TC1]f sinfo PARTITION AVAIL TIMELIMIT NODES STATE NODELIST SCSEGEU_UG up infinite 3 mix SCSEGEU-TC1-[01-03] SCSEGEU_UG up infinite 4 idle SCSEGEU-TC1-[04-07] SCSEGEU_MSAI up infinite 4 idle SCSEGEU-TC1-[04-07] SCSEGEU_MSAI up infinite 4 mix SCSEGEU-TC1-[04-07] SCSEGEU_MSAI up infinite 50,000EllsT has been down # To view our own job history > sacctformat=user,jobid,jobname%+15,qos,alloctres%+50,nodellst,start,elapsed,state,reason%+20 # Job viewing, excluding those lines with "batch" > sacct -a-format=user,jobid,jobname%+15,qos,alloctres%+50,nodellst,start,elapsed,state,reason%+20 grep-wv batch # List the jobs executed by specific user <username> &gt; sbatch job script to the job queue &gt; sbatch job script to the job queue &gt;</username>		CfgTRES=cpu=64,mem=128508M,billing=64,gres/gpu=8 AllocTRES=cpu=26.mem=63480M.gres/gpu=7
CurrentWatts=0       AveNatts=0         ExtSensorsJoules=n/s       ExtSensorsTemp=n/s         You can reference the resources status of the node, based on the information at the lines "CfgTRES" and "AllocTRES". The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         sinfo       # Display state of the compute nodes in respective partition >> sinfo         [scctest66SCSEGEU-TC1]# sinfo         [scctest66SCSEGEU_TC1]# sinfo         [scctest66SCSEGEU_UG         up infinite       3         mix       SCSEGEU_UC=04         up infinite       4         idle       SCSEGEU_TC1=[01-03]         SCSEGEU_UG       up infinite         4       idle       SCSEGEU_TC1=[04-07]         SCSEGEU_MSAI       up infinite       4         for STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down         sacct       # To view your own job history         >> sacctformat=jobid.jobname*+15.qos,alloctres*+50,nodelist,start,elapsed,state,reason*+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid.jobname*+15.qos,alloctres*+50,nodelist,start,elapsed,state,reason*+20         grep -wv batch         # List the jobs executed by specifi		CapWatts=n/a
You can reference the resource status of the node, based on the information at the lines "CfgTRES" and "AllocTRES". The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         sinfo       # Display state of the compute nodes in respective partition         >> sinfo       [scetest6@SCSEGEPU-TC1] # sinfo         PARTITION       AVAIL       TIMELIMIT         NODES       STATE       NODELIST         SCSEGEPU_UG       up       infinite       3       mix       SCSEGEPU-TC1-[01-03]         SCSEGEPU_UG       up       infinite       4       idle       SCSEGEPU-TC1-[04-07]         SCSEGEPU_MSAI       up       infinite       4       mix       SCSEGEPU-TC1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGEPU-TC1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGEPU-TC1-[08-11]         For       STATE not showing "idle" or "mix", may indicate the node under NODELIST		CurrentWatts=0 AveWatts=0 ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.         # Display state of the compute nodes in respective partition         >> sinfo         [scetest6@SCSEGFU-TC1] # sinfo         PARTITION       AVAIL         SCSEGFU_UG       up infinite         3       mix         SCSEGFU_UG       up infinite         4       idle         SCSEGFU_MSAI       up infinite         4       mix         SCSEGPU_MSAI       up infinite         4       mix         SCSEGPU_MSAI       up infinite         4       mix         SCSEGPU_MSAI       up infinite         5       scct         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down         scact       # To view your own job history         >> sact       # To view all job history for the day, option "-a"         >> sact       -a -format=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20		You can reference the resource status of the node, based on the information at the lines "CfaTRES" and "AllocTRES".
sinfo       # Display state of the compute nodes in respective partition         >> sinfo       [scetest6@SCSEGPU-TC1] # sinfo         PARTITION       AVAIL       TIMELIMIT         NODES       STATE       NODELIST         SCSEGPU_UG       up       infinite       3       mix       SCSEGPU-TC1-[01-03]         SCSEGPU_UG       up       infinite       4       idle       SCSEGPU-TC1-[04-07]         SCSEGPU_MSAIL       up       infinite       4       mix       SCSEGPU-TC1-[04-07]         SCSEGPU_MSAIL       up       infinite       note       mix       SCSEGPU-TC1-[04-07]         SCSEGPU_MSAIL <t< th=""><th></th><th>The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.</th></t<>		The values shown under "AllocTRES" are the resources being allocated to the running jobs in the node.
<pre>&gt;&gt; sinfo [scetest6@SCSEGPU-TC1]# sinfo PARTITION AVAIL TIMELIMIT NODES STATE NODELIST SCSEGPU_UG up infinite 3 mix SCSEGPU-TC1-[01-03] SCSEGPU_UG up infinite 4 idle SCSEGPU-TC1-[04-07] SCSEGPU_MSAI up infinite 4 mix SCSEGPU-TC1-[08-11] For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down sacct # To view your own job history &gt;&gt; sacctformat=jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # To view all job history for the day, option "-a" &gt;&gt; sacct -aformat=user,jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # Job viewing, excluding those lines with "batch" &gt;&gt; sacct -aformat=user,jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # List the jobs executed by specific user <username> &gt;&gt; sacct -u <username> &gt;&gt; sacct -u <username> &gt;&gt; sacct -u <username> &gt;&gt; sacct +format=user,jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # List the jobs executed by specific user <username> &gt;&gt; sacct -aformat=user,jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # List the jobs executed by specific user <username> &gt;&gt; sacct -u <username> &gt;&gt; sacct -jobid.&gt;&gt; scance  # The jobid can be obtained by the command "squeue" # Cancel a job base on its jobid &gt;&gt; scancel <jobid></jobid></username></username></username></username></username></username></username></username></username></username></username></pre>	sinfo	# Display state of the compute nodes in respective partition
iscetest6@SCSEGPU-TC1]# sinfo         PARTITION       AVAIL         SCSEGPU_UG       up         infinite       3         mix       SCSEGPU_TC1-[01-03]         SCSEGPU_UG       up         infinite       4         idle       SCSEGPU_TC1-[04-07]         SCSEGPU_MSAI       up         infinite       4         mix       SCSEGPU_TC1-[08-11]         For STATE not showing "de" or "mix", may indicate the node under NODELIST has been down         sacct       # To view our own job history         >> sacctformat=jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch       # List the jobs executed by specific user <username>         &gt;&gt; sacct -u       u</username>		>> sinfo
PARTITION       AVAIL       TIMELIMIT       NODES       STATE NODELIST         SCSEGPU_UG       up       infinite       3       mix       SCSEGPU-TC1-[01-03]         SCSEGPU_MSAI       up       infinite       4       idle       SCSEGPU-TC1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGPU-TC1-[08-11]         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down       scacet       # To view your own job history         >> sacct       # To view your own job history       >> sacctformat=jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"       >> sacct -aformat=user,jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"       >> sacct -aformat=user,jobid.jobname%+15.qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch       # List the jobs executed by specific user <username>       &gt;&gt; sacct -u       <username>       &gt;&gt; sacct -u       <username>       &gt;&gt; sacct -u       <username>       &gt;&gt; sbatch       # Submit a job script to the job queue       &gt;&gt; sbatch job.sh        &gt;&gt; scancel &lt; job base on its jobid</username></username></username></username>		[scetest6@SCSEGPU-TC1]# sinfo
SCSEGPU_UG       up       infinite       3       mix       SCSEGPU_C1-[01-03]         SCSEGPU_MSAI       up       infinite       4       idle       SCSEGPU_C1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGPU_C1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGPU_C0-[08-11]         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down       sacct       # To view your own job history         sacctformat=jobid.jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20       # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20       # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20       grep -wv batch         # List the jobs executed by specific user <username>       &gt;&gt; sacct -u <username>         &gt;&gt; sacct -u <username>       &gt;&gt; sacct -u <username>         &gt;&gt; sacct -u <username>       &gt;&gt; sbatch         # The jobid can be obtained by the command "squeue"       # Cancel a job base on its jobid         &gt;&gt; scancel       -jobid&gt;</username></username></username></username></username>		PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
SCSEGPU_UG       up       infinite       4       idle       SCSEGPU-TC1-[04-07]         SCSEGPU_MSAI       up       infinite       4       mix       SCSEGPU-TC1-[08-11]         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down       ************************************		SCSEGPU_UG up infinite 3 mix SCSEGPU-TC1-[01-03]
SCSEGFU_MSAI       up infinite       4 mix SCSEGFU_TC1-[08-11]         For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down         sacct       # To view your own job history         >> sacctformat=jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>		SCSEGPU_UG up infinite 4 idle SCSEGPU-TC1-[04-07]
For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down         sacct       # To view your own job history         >> sacctformat=jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid</username></username>		SCSEGPU_MSAI up infinite 4 mix SCSEGPU-TC1-[08-11]
sacct       # To view your own job history         >> sacctformat=jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid</username></username>		For STATE not showing "idle" or "mix", may indicate the node under NODELIST has been down
>> sacctformat=jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>	sacct	# To view your own job history
# To view all job history for the day, option "-a"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         # Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>		>> sacctformat=jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20
<pre>&gt;&gt; sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 # Job viewing, excluding those lines with "batch" &gt;&gt; sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 grep -wv batch # List the jobs executed by specific user <username> &gt;&gt; sacct -u <username> sbatch # Submit a job script to the job queue &gt;&gt; sbatch job.sh scancel # The jobid can be obtained by the command "squeue" # Cancel a job base on its jobid &gt;&gt; scancel <jobid></jobid></username></username></pre>		# To view all job history for the day, option "-a"
<pre># Job viewing, excluding those lines with "batch" # Job viewing, excluding those lines with "batch" &gt;&gt; sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 grep -wv batch # List the jobs executed by specific user <username> &gt;&gt; sacct -u <username> sbatch # Submit a job script to the job queue &gt;&gt; sbatch job.sh scancel # The jobid can be obtained by the command "squeue" # Cancel a job base on its jobid &gt;&gt; scancel <jobid></jobid></username></username></pre>		>> sacct -aformat=user,jobid.jobname%+15.gos,alloctres%+50,nodelist.start,elapsed,state,reason%+20
# Job viewing, excluding those lines with "batch"         >> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20         grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch         # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>		
<pre>&gt;&gt; sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20 grep -wv batch # List the jobs executed by specific user <username> &gt;&gt; sacct -u <username> sbatch # Submit a job script to the job queue &gt;&gt; sbatch job.sh scancel # The jobid can be obtained by the command "squeue" # Cancel a job base on its jobid &gt;&gt; scancel <jobid></jobid></username></username></pre>		# Job viewing, excluding those lines with "batch"
grep -wv batch         # List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>		>> sacct -aformat=user,jobid,jobname%+15,qos,alloctres%+50,nodelist,start,elapsed,state,reason%+20
# List the jobs executed by specific user <username>         &gt;&gt; sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username></username>		grep -wv batch
share the jobs encented by specific user (username)         >> sacct -u <username>         sbatch       # Submit a job script to the job queue         &gt;&gt; sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         &gt;&gt; scancel <jobid></jobid></username>		# List the jobs executed by specific user <username></username>
sbatch       # Submit a job script to the job queue         >> sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         >> scancel <jobid></jobid>		
statistical job script to the job queue         >> sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         >> scancel <jobid></jobid>	chatch	# Submit a job script to the job quare
>> sbatch job.sh         scancel       # The jobid can be obtained by the command "squeue"         # Cancel a job base on its jobid         >> scancel <jobid></jobid>	spatch	# SUDMILLA JOB SCRIPT TO THE JOB QUEUE
scancel # The jobid can be obtained by the command "squeue" # Cancel a job base on its jobid >> scancel <jobid></jobid>		>> sbatch job.sh
# Cancel a job base on its jobid >> scancel <jobid></jobid>	scancel	# The jobid can be obtained by the command "squeue"
>> scancel <jobid></jobid>		# Cancel a job base on its jobid
I		>> scancel <jobid></jobid>
		1

squeue	# Display the jobs in the job queue
	>> squeue
	# List more available information with option 'la' >> squeue -la
	# List the running jobs submitted by the user <username></username>
	>> squeue -u <username></username>
seff	# Slurm Job Efficiency Report
	# Display the statistics of resource being utilized by the completed job, allow to review on the resource
	assignment for the job
	>> seff <jobid completed="" job="" of=""></jobid>

## Guideline for SLURM job submission

#### 1. Only apply the QoS assigned to you

The job submitted with unauthorized QoS (QoS not assigned for your use) will be terminated by the system with no prior notification.

2. Apply the QoS resources within the limit in your job script

You may use cu	ustomized	script/command to view	your account and (	QoS Info	
>> mytcinfo					
C[scetest6@SCS User ] uid=30902(sce Home Director =====User	SEGPU-TC1: Info of so etest6) gi ry = /home Info of s	~] # mytcinfo etest6 in TC Cluster d=30902(scetest6) group c/UG/scetest6 ccetest6 in SLURM DB====	os=30902(scetest6	 ),17184921(tclus	sers)
= Us	ser	Cluster		QOS	
scetes	st6 scs	e-gpu-clusterl		normal	
===QOS Info== Na	ame	MaxTRESE	PU MaxJobsP	U MaxWal	
norr Partitic	nal c on	pu=20,gres/gpu=1,mem=64	IG	2 06:00:0	00
SCSEGPU_UG					
In your job scri	pt, ensure	to apply the job flag with	stated value below	v or equal to (<=)	the amount in the assigned QoS.
MaxJobsPU	The max You may	mum of number of jobs a submit two jobs to run at	llow to run at a tin <i>a time</i>	ne <= <b>2</b>	
MaxWall	Total cor	nputation of your running	jobs must not exc	eed the maximur	n wall time <= 6-hr

MaxTRESPUTotal amount of resources allocated in your two job scripts must not exceed the maximum values.For example: No of CPU/cores <=20, Memory <=64GB and No of GPU Card=1, as stated in QoS assigned.</td>

# Other customized scripts/commands are available under /tc1share/tc-scripts #

#### 3. Create/Edit your SLURM job script

You may use any text editor available at your host machine, to create the job script, ensuring the job script is saved with file extension "**.sh**". Then, transfer to your home directory in the GPU Cluster to execute.

	« z CWsupport » Z-scripts » test	× ē.	Search test	Q
	·	• 0	ocorer test	F
Organize 🔻 🛛 Ne	w folder			
1.0.11	^ Name	Туре	Date modified	Size
Quick access Creative Cloud	job1	SH File	24/11/2021 1:49 PM	N
📃 This PC				
HP_S231d (E:)	v <			>
File name:	job.sh			~
Save as type:	All Files			~
Hide Folders	Encoding: A	ANSI ~	Save Canc	cel
the Head N	ode, you may use those lin	ux command line	editors to edit you	ur job script: <i>vi, nano</i> or <i>ne</i>
the reference	25 miles on Linux Command-II	ne cuitors:		
tps://www.gun	ksforgeeks.org/nano-text-editor-	in-linux/		

The following a	are the basic job flags to apply	y in the job script:
Resource	Flag Syntax	Description
Partition*	partition=SCSEGPU_UG	# Specify the node group for your job execution, defines as <i>Partition</i> # Command to list the available Partitions and allowed QoS in the Cluster >> scontrol show partition
QoS*	qos=normal	# Specify the QoS to apply for the job # The QoS must be one of them allow to execute in the Partition
nodes*	*nodes=1	# Number of compute nodes for the job # Max node allows per job: 1
	nodelist= <hostname></hostname>	# Optional: you may specify the node (hostname) to run the job # Command to list the idle nodes and their hostname in the Partition >> sinfo -N <name of="" partition="">   grep idle &gt;&gt; sinfo -N SCSEGPU_UG   grep idle</name>
GPU*	gres=gpu:1	<ul><li># Specify the use of GPUs on compute nodes.</li><li># The number of GPU card to use is corresponding to the assigned QoS</li><li># This option must be present in the job script, for the system to deploy GPU resource</li></ul>
Memory*	mem=8000M Or mem=8G	<ul> <li># Specify on the memory to apply for the job.</li> <li># The value is corresponding to the assigned QOS, must not exceed the assigned maximum memory size.</li> <li># This option must be present in the job script, for the system to deploy the memory for the computation.</li> </ul>
CPUs/cores	ntasks-per-node=2	<ul> <li># Optional: only add this flag if want to apply more than one core for the computation</li> <li># Specify the number of "tasks" (cores) per node, for use with distributed parallelism. Default value=1</li> </ul>
	cpus-per-tasks=2	<ul> <li># Optional: only add this flag if want to apply more than one core for the computation</li> <li># Specify the number of CPU-cores to allocate to per task, for use with shared memory parallelism. Default value=1</li> </ul>
Job name*	job-name=MyJob	# Name of job
Output file*	output=output_%j.out Or output=output_%x_%j.out	# State the name of the file for standard output Filename patterns: %x: job name %j: job id, generated by SLRUM
Error file*	error=error_%j.err Or error=error_%x_%j.err	# State the name of the file for error log – if any
time	time=01:00:00	<ul> <li># Specify a time limit for the job – hh:mm:ss</li> <li># The max timing is corresponded to <i>MaxWall</i> in assigned QOS</li> <li># Optional, only input if requires limiting the job timing</li> </ul>

The flags with \* must specify in the job script

The following is an example of a standard job script: job.sh

<pre>#!/bin/bash #SBATCHpartition=SCSEGPU_UG #SBATCHqos=normal #SBATCHnodes=1 #SBATCHgres=gpu:1 #SBATCHmem=8G #SBATCHjob-name=MyJob #SBATCHoutput=output_%x_%j.out #SBATCHerror=error_%x_%j.err</pre>	Remarks:         You may add necessary flags from the table above.         Do note that if you exceed the limit for your QOS,         the job will not run.         From this job script, 2 files will be created:         output_%x_%j.out         > The standard output from running the code will be saved here         error_%x_%j.err         > Error log from job if any
module load anaconda	Load necessary modules needed to run the code
source activate TestEnv	Activate your conda environment (if any)
python test.py	Run your code

Update [ 19 January 2023 ]

#### 4. Submit your job script

Action to submit a job	Command
To submit your job script, where job.sh is the name of your job script	sbatch job.sh
The system should respond with a job ID:	
[scetest6@SCSEGPU-TC1 sample-1]\$ sbatch ; ASubmitted batch job 34483	job.sh

#### 5. Avoid using the command "srun" to submit job

The command "*srun*" is to submit job at the command line for *real time execution*. You have to maintain your SSH session until the whole process completed. The disconnection of your SSH session may kill the process and causing you to lose the control over the execution. Thus, for the jobs requiring more than an hour to compute, are advised to submit using the command "*sbatch*".

The command "*sbatch*" is to submit job for later execution, handling by SLURM in the background. Once submitted the job using "*sbatch*", you may exist from your SSH session, with no consequence.

To avoid high volume of SSH connections to the Cluster Head Node, all users are advised to use the command "*sbatch*" for job submission.

#### 6. Verify on the job queue and node operational status

Command to verify on the job queue:
>> squeue
Command to verify on the resources being in used by the running jobs >> scontrol show node SCSEGPU-TC1-[01-07]
# Look at the overall resources being allocated for node operation and running jobs
NodeName=SCSEGPU-TC1-11 Arch=x86 64 CoresPerSocket=16
CPUAlloc=22 CPUTot=64 CPULoad=34.48
AvailableFeatures=(null)
ActiveFeatures=(null)
Gres=gpu:8(S:0-1)
NodeAddr=SCSEGPU-TC1-11 NodeHostName=SCSEGPU-TC1-11 Version=19.05.5
OS=Linux 3.10.0-1062.1.1.e1/.x86_64 #1 SMP Fr1 Sep 13 22:55:44 UIC 2019 DealMemorys128508 AllocMeme103488 FreeMeme75357 Socketes2 Boardes1
State=MUSFD ThreadsDerCore=2 TmpDisk=0 Weight=1 Owner=N/A MCS label=N/A
Partitions=SCSEGPU MSAI
BootTime=2021-12-22T09:14:22 SlurmdStartTime=2021-12-30T13:48:43
CfgTRES=cpu=64,mem=128508M,billing=64,gres/gpu=8
AllocTRES=cpu=22,mem=103488M,gres/gpu=8
CapWatts=n/a
CurrentWatts=0 AveWatts=0
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s

- The node selected to run the job is based on the available resources (CPU, Memory, GPU and Compute nodes) at the time of submission
- For the jobs failed the assignment will go into PENDING queue
- > The running jobs terminated due to node reset or failure, will also be appearing in the PENDING queue

#### # How to spot failing node #

- For node with state showing "DRAIN", indicating it has failed task. This node is alive but may not able to accept new job and the performance of existing running jobs may also be affected.
- Requiring the administrator to reset the affected node for operation. The impact of the reset is the existing running jobs in the affected node will be terminated. The affected jobs will have to join the PENDING queue for reassignment.

Update [ 19 January 2023 ]



- Common reasons showing for such affected jobs in the PENDING queue: *Priority, Resources*
- The job will be executed when there is ready node and resource to assign within Time Limit (referenced to the maximum wall time (MaxWall) stated in the QoS)
- > The pending job will be removed by the system when exceeded the Time Limit

#### Advices:

If there are many jobs in the PENDING queue with reasons of "Priority" or "Resources", please defer your job submission until those jobs are cleared in the queue

#### 7. Process your submitted SLURM job

Actions to check on job status	Command
Display the jobs in the scheduling queue	squeue -la
Display job history for a user <username></username>	sacct -u <username></username>
Show the detail for a running/pending job <jobid></jobid>	scontrol show jobid <jobid></jobid>

If the Status indicates PD (PENDING) and NODELIST (REASON) indicates a <u>reason in bracket</u>, do note that the job has failed to run. Please cancel the job if this happens.

[scetest6@SCSEGPU-7	[C1 ~]\$ squ	eue					
JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST (REASON)
263	SCSEGPU	TestJob	scetest6	PD	0:00	1	(QOSMaxJobsPerUserLimit)

For submitted job failed the QoS limit check (*exceeded any of the values stated for the QoS*), will go into the PENDING queue

Common reasons for job associates to QoS failure:

- QOSMaxJobsPerUserLimit
  - Verify on your total number of jobs in Running queue, must not exceed the value stated in "MaxJobsPU"
  - Extra job will go into the PENDING queue
- QOSMaxMemoryPerUser, QOSMaxCpuPerUserLimit, QOSMaxGRESPerUser
  - The total resource count [ **#SBATCH --mem=x**, **#SBATCH --ntasks-per-node=x** ] stated in all your submitted job scripts using the same QoS, must not exceed the max value stated in **"MaxTRESPU**"
- QOSMaxGRESPerUser
  - This error may appear if unable to assign your jobs using the same QoS, to the allocated (1) GPU Card
  - For this case, restrict to submit only one (1) job to run at a time

The job will only be executed when passed the checks for QoS limit and node assignment within the Time Limit (referenced to **MaxWall** stated in the QoS)

The pending job will be removed by the system when reached the Time Limit.

Advice for such encounter:

Cancel your pending jobs failed the QoS limit Check

Update [ 19 January 2023 ]

Page 18 of 25

<ul> <li>Review and modify the resource values in you</li> <li>Restrict your number of job submission</li> </ul>	ır job script
Action to cancel the job	Command
To cancel a job, where 263 is your job id	scancel 263
Once the job has finished running, it will be remove not specify a specific path for the output file in you the SLURM job.	/ed from the queue. You should be able to see the output file. If you ur SLURM script, it will be created in the directory where you subm
Error MyJob 34483.err job.sh Out [scetest6@SCSEGPU-TCl sample-1]\$	put_MyJob_34483.out test.py
o know more about your submitted jobs	
You may use the customised script to view your jo	b history for the day, with details on applied QoS, allocated resource
Sector (1997) (199 ) (19977) (19977) (1997) (19977) (1997) (1997) (1997) (1997) (19	gnea, joo-saonni tinne, start tinne, elapsea tinne, state ana reason
<pre>[scetest68SCSEGPU-TC1:~/jobscript]# myjobhistory -List Running Jobs User Labid Labiana L ges LableComputes Latart Laborat L</pre>	
<pre>y User   jobla   jobname   qos   allockesources   start   elapsed   scetest6 65897 runl normal billing=1, scetest6 65898 runl normal billing=1,</pre>	nodelist   state   reason cpu=l,gres/gpu=l,mem=10G,node=1 2022-12-14T13:10:57 00:25:18 SCSEGFU-TC1-01 COMP cpu=l,gres/gpu=l,mem=10G,node=1 2022-12-14T13:42:48 00:21:12 SCSEGFU-TC1-01 CANCE
	Perminentiale #
# Know the actual GPO Cara being assigned for your second second and "scontrol", with option "-d"	our kunning job #
>> scontrol -d show jobid <jobid job<="" of="" running="" td=""><td>&gt;</td></jobid>	>
Look for the line "JOB GRES=qpu:X	
NodeList=SCSEGPU-TC1-01 BatchHost=SCSEGPU-TC1-01	
NumNodes=1 NumCPUs=10 NumTasks=10 CPUs	s/Task=1 ReqB:S:C:T=0:0:*:*
<pre>TRES=cpu=10,mem=32G,node=1,billing=10, Socks/Node=* NtasksPerN:B:S:C=10:0:*:*</pre>	,gres/gpu=1 * CoreSpec=*
JOB_GRES=gpu:1	
Nodes=SCSEGPU-TC1-01 CPU_IDs=0-9 Mer MinCPUsNode=10 MinMemoryNode=32G MinTr	n=32768 GRES=gpu:1(IDX:0) mpDiskNode=0
Features=(null) DelayBoot=00:00:00	
OverSubscribe=OK Contiguous=O Licenses	<pre>B= (null) Network= (null)</pre>
# Know the CPU and Memory being utilized for yo	our Completed Job #
Command to display the resource utilization statis	stics for your completed job, let you review and deploy the right nu
of resources (LPU/cores and Memory) for your cor >> seff <ioh id=""></ioh>	nputation
[scetest6@SCSEGPU-TC1:~/jobscript]# seff @	55897
Job ID: 65897	
Cluster, cose mu cluster]	
Cluster: scse-gpu-clusterl User/Group: scetest6/scetest6	
Cluster: scse-gpu-clusterl User/Group: scetest6/scetest6 State: COMPLETED (exit code 0) Cores: l	
Cluster: scse-gpu-clusterl User/Group: scetest6/scetest6 State: COMPLETED (exit code 0) Cores: 1 CPU Utilized: 00:00:06 CPU Utilized: 00:00:06	ltime
Cluster: scse-gpu-cluster1 User/Group: scetest6/scetest6 State: COMPLETED (exit code 0) Cores: 1 CPU Utilized: 00:00:06 CPU Efficiency: 0.40% of 00:25:18 core-wal Job Wall-clock time: 00:25:18	lltime
Cluster: scse-gpu-clusterl User/Group: scetest6/scetest6 State: COMPLETED (exit code 0) Cores: 1 CPU Utilized: 00:00:06 CPU Efficiency: 0.40% of 00:25:18 core-wal Job Wall-clock time: 00:25:18 Memory Utilized: 74.91 MB Memory Utilized: 74.91 MB	lltime

# Notice for Specific Coursework

For CE4042/CZ4042, there are ready shared conda environments created for your assignment:

/apps/conda_env/CZ4042_v4	/apps/conda_env/CZ4042_v3	/apps/conda_env/CZ4042_v2
Apps installed: Python 3.10 Tensorflow 2.9 Cudatoolkit 11.7	Apps installed: Python 3.8 Tensorflow 2.6 Cudatoolkit 11.3	Apps installed: Python 3.9 Tensorflow 2.6 Cudatoolkit 11.3
Environment Variable (For Tensorflow 2.9) TF_ENABLE_ONEDNN_OPTS=0		

To load Anaconda and activate your shared environment, simply type the following:

- module load anaconda
- conda activate /apps/conda\_env/CZ4042\_v4

```
[scetest6@SCSEGPU-TC1:~] # module load anaconda
[scetest6@SCSEGPU-TC1:~] # module list
Currently Loaded Modulefiles:
1) anaconda
[scetest6@SCSEGPU-TC1:~] # conda activate /apps/conda_env/CZ4042_v4
(/apps/conda_env/CZ4042_v4) [scetest6@SCSEGPU-TC1:~] # python -V
Python 3.10.4
(/apps/conda_env/CZ4042_v4) [scetest6@SCSEGPU-TC1:~] # conda list | grep cudatoolkit
udatoolkit
                         11.7.0
                                             hd8887f6_10 nvidia
(/apps/conda_env/C24042_v4) [scetest6@SCSEGPU-TC1:~] # conda env config vars list
TF ENABLE ONEDNN OPTS = 0
(/apps/conda_env/CZ4042_v4) [scetest6@SCSEGPU-TC1:~] # python -c 'import tensorflow as tf; print(tf._version_)'
2.9.1
(/apps/conda_env/CZ4042_v4) [scetest6@SCSEGPU-TC1:~]#
```

# Miscellaneous

# **SSH** Tunneling

The GPU Cluster is only accessible via SSH connection. For the user requiring running *Jupyter Notebooks*, must learn to setup the SSH tunnel for computation.

SSH tunneling (also known as SSH port forwarding) is a method of creating an encrypted SSH connection between a client and a server through which services port can be relayed.

The following is an exemplary guide:

Running Jupyter Notebook on a computer node that needs to be contacted by the web browser on your local computer

1. SSH to GPU Head Node and setup Jupyter Notebook in your home directory

Description	Execution
Load the anaconda module	# Execute Command
	>> module load anaconda
Create and activate a conda	# Execute Command
environment	>> conda create -n RunJupyter
	>> conda activate RunJupyter
Install the required package for	# Execute Command
Jupyter Notebook from third-party	>> conda install -c conda-forge notebook
channel " <i>conda-forge</i> "	>> conda install -c conda-forge nb_conda_kernels
	>> conda install -c conda-forge jupyter_contrib_nbextensions
Deactivate the conda environment	# Execute Command
	>> conda deactivate

#### 2. Create the job script

Description	Execution
Create the job script to run Jupyter	# Sample of the job script
Notebook. For this example, the job script is named as " <i>run1.sh</i> "	#!/bin/sh #SBATCHpartition=SCSEGPU_UG #SBATCHqos=normal #SBATCHgres=gpu:1 #SBATCHnodes=1
	#SBATCHmem=10G #SBATCHjob-name=run1 #SBATCHoutput=output_%x_%j.out #SBATCHerror=error_%x_%j.err
	module load anaconda source activate RunJupyter jupyter-notebookip=\$(hostname -i)port=8887
Submit the job script	# Execute Command >> sbatch run1.sh

### 3. Verify the access information in the error log.

Take note of the *IP address, port number* and the *URL* for web access

```
[scetest6@SCSEGPU-TC1:~/jobscript]# 11
total 33
-rw-rw-r-- 1 scetest6 scetest6 1191 Dec 14 13:11 error run1 65897.err
-rw-rw-r-- 1 scetest6 scetest6 277 Dec 14 13:11 output_run1_65897.out
-rw-rw-r-- 1 scetest6 scetest6 308 Dec 14 13:10 runj.sh
[scetest6@SCSEGPU-TC1:~/jobscript] # more output runl 65897.out
[scetest6@SCSEGPU-TC1:~/jobscript] # more error run1 65897.err
[I 13:11:04.187 NotebookApp] [nb_conda_kernels] enabled, 2 kernels found
[I 13:11:04.391 NotebookApp] Writing notebook server cookie secret to /home/UG/scetest6/.1
[I 13:11:05.506 NotebookApp] [jupyter_nbextensions_configurator] enabled 0.6.1
[I 13:11:05.509 NotebookApp] Serving notebooks from local directory: /home/UG/scetest6/job
[I 13:11:05.509 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[I 13:11:05.509 NotebookApp] http://10.128.10.11:8887/?token=af4d500318edf22b5e314dfc066f7
[I 13:11:05.509 NotebookApp] or http://127.0.0.1:8887/?token=af4d500318edf22b5e314dfc066f
[I 13:11:05.509 NotebookApp] Use Control-C to stop this server and shut down all kernels
[C 13:11:05.530 NotebookApp]
    To access the notebook, open this file in a browser:
        file:///home/UG/scetest6/.local/share/jupyter/runtime/nbserver-45528-open.html
    Or copy and paste one of these URLs:
        http://10.128.10.11:8887/?token=af4d500318edf22b5e314dfc066f7fflcf880e474fbb1222
     or http://127.0.0.1:8887/?token=af4d500318edf22b5e314dfc066f7ff1cf880e474fbb1222
```

4. In your local machine, start another SSH session to SCSEGPU-TC1 Cluster:

🕵 PuTTY Configuration		? ×
Category: 	Basic options for your Puil         Specify the destination you want to         Host Name (or IP address)         172.21.148.240         Connection type: <ul> <li>SSH</li> <li>Serial</li> <li>Other:</li> </ul> Load, save or delete a stored sessi         Saved Sessions         scsegpu+c1.scse.ntu.edu.sg         Default Settings         scsegpu+c1.scse.ntu.edu.sg         Close window on exit:         Always       Never	TTY session connect to Port 22 Telnet ~ on Load Save Delete
<u>A</u> bout <u>H</u> e	p <u>O</u> pen	Cancel

Update [19 January 2023]

Page **22** of **25** 

### 5. Expand the category of "SSH" and select "Tunnels"

🕵 PuTTY Configuration			?	×
Category:				
Keyboard     Bell     Features     Vindow     Appearance     Behaviour     Translation     Selection     Colours	Options c Port forwarding Local ports acc Remote ports d Envanded ports L8887 10.128	ontrolling SSH por cept connections f lo the same (SSH 3.10.11:8887	t forwarding from other hosts -2 only) Remov	c
Connection     Data     Froxy     Or SSH     Frox Kex     West Keys     Or Cipher     Other     Other	Add new forwarde Source port Destination Local Auto	d port: 8887 10.128.10.11:88 O Remote O IPv4	Add 87 O Dynamic O IPv6	
About TTY TTY X11 Tunnels Bugs Help		Open	Cance	4

Input the IP address and port number obtained in the Error Log as show above. Click on the button **"Add"** to add into the Forwarded ports

- 6. Click on the button "*Open*" to start the SSH session. Login using your NTU username and password to establish the connection.
- 7. Finally, you may access the URL in your local machine (Copy and paste one of the URLs to your browser for access)

To access the notebook, open this file in a browser: file:///home/UG/scetest6/.local/share/jupyter/runtime/nbs	server-4552	8-open.	html	
Or copy and paste one of these URLs: http://10.128.10.11:8887/?token=af4d500318edf22b5e314dfc0 or http://127.0.0.1:8887/2token=af4d500318edf22b5e314dfc066d	066f7fflcf8	80e474f	bb12	22
	1/11/01/00/08	1/11001		-
C Home Page - Select or create a X +			-	
← → C O D http://127.0.0.1:8887/tree		☆		Ŧ
📁 Jupyter		Quit	Logout	]
Files Running Clusters Nbextensions				
Select items to perform actions on them.		Upload	New •	0
	Name 🕹	Last Modified	File size	
C error_run1_66897.err		seconds ago	1.43 k	в
O output_run1_65897.out		24 minutes ago	277	в
🗖 🗅 runj.sh		24 minutes ago	308	в

8. The operation of Jupyter will terminate when you click on the button "*Quit*" in the web page. Then, you may logout from the SSH session set with the forwarded port.

Update [ 19 January 2023 ] Pa	ge <b>23</b> of <b>25</b>
-------------------------------	---------------------------

# Other Resources for application

### QoS

The user may apply additional QoS for...

- Ionger computation hour, selecting range: 8-hr, 12-hr, 24-hr, till max 48-hr
- computation with more resource allocation: CPUs, Memory and GPU Card

Usage period: 1-month

Method to apply:

- Must provide the reason for the usage and verification (coding and computation result executed in the cluster) on the assigned QoS unable to support the user's operation.
- Send the request e-mail to SCSEGPU-TC-Support [scsegpu-tc@ntu.edu.sg]
- > The administrator will evaluate, verify on the real requirement, and then assign the QoS accordingly
- > The requestor will be notified, upon successful application

### **Additional Storage Space**

The user may apply additional storage space for computation data more than 100GB A share folder will be created in another storage to assign to the user.

Usage period: 4-months

Maximum usage quota: depending on the available free space and approval

Method to apply:

- Send the request e-mail to SCSEGPU-TC-Support [scsegpu-tc@ntu.edu.sg]
- Must provide the reason and verification for the usage request
- > The user will be receiving the PFD form, via e-mail, to complete for the application
- > The user will be notified of the resource assignment, upon successful application

### **Terms and Conditions**

- > The application is subjecting to resource availability, cluster operational workload and administration approval
- > The usage of the assigned resource will be monitored
- The administrator claims the right to revoke the assigned resource if the usage has been verified underuse or impacting on the operation of the GPU Cluster
- > The assigned resource will be removed upon usage expiry, with no prior notification

# **Important Notice**

- 1. There is NO backup provided for data stored in your home directory. You are responsible for protecting and maintaining your own data.
- 2. Resources are to be used for academic purposes only. Use of resources for a purpose other than that for which they were granted will result in the termination of account.
- 3. Your computation is being monitored. For jobs submitted with Unauthorised QoS (NOT assigned for your use) and operation, will be terminated with no prior notification. Severe offender may be barred from the service.
- 4. DO NOT access directories other than your Home Directory or Shared folder. Unauthorised access to other directories or files, even for the purpose of "browsing", shall be deemed as a security breach. Serious action will be taken against users committing any breach to the SCSE GPU Cluster.
- 5. For issue regarding your coursework and project, please consult your Supervisor or TA (Teaching Assistant)
- 6. For technical issue, drop an email to SCSEGPU-TC-Support [scsegpu-tc@ntu.edu.sg]

~ End ~