CLOUDAM-HPC User Manual

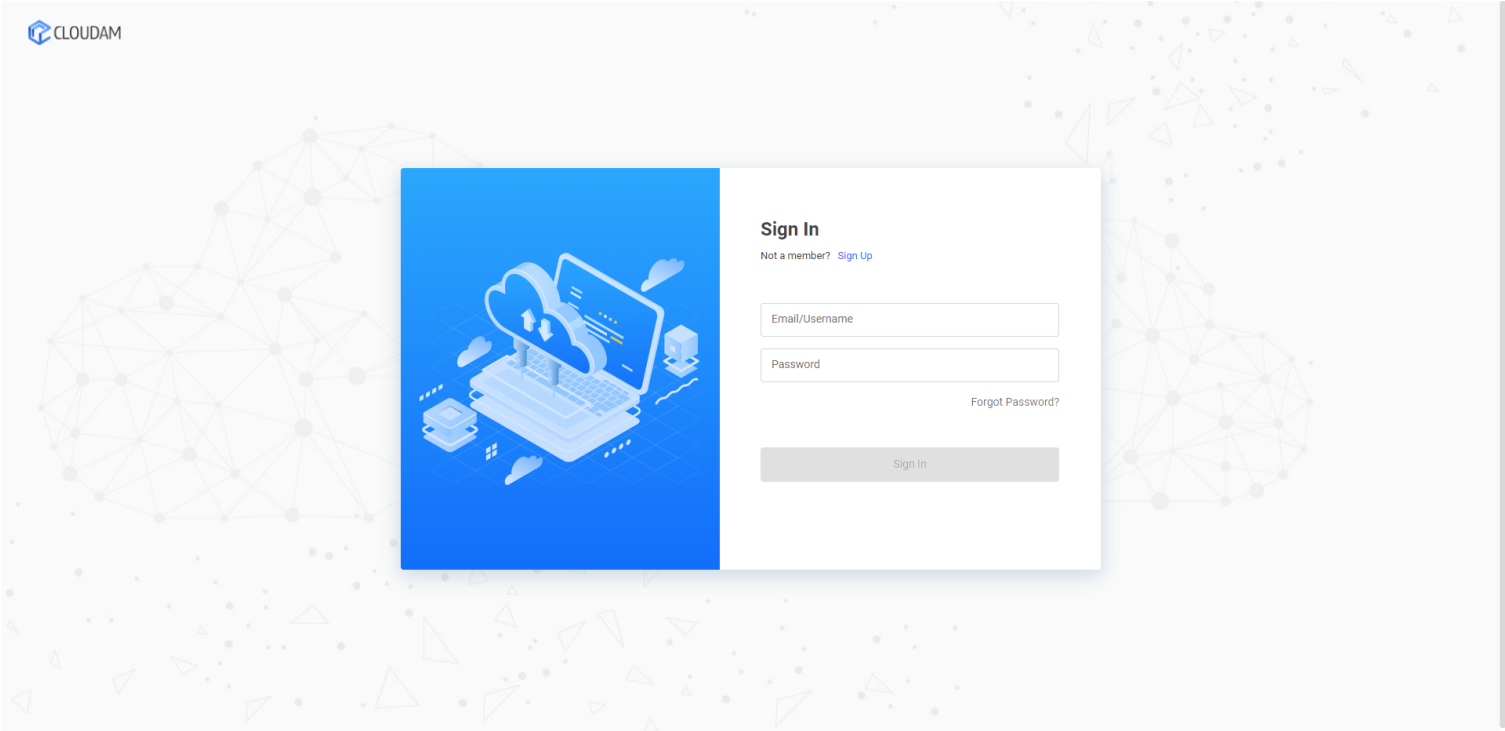
Table of contents:

Log in

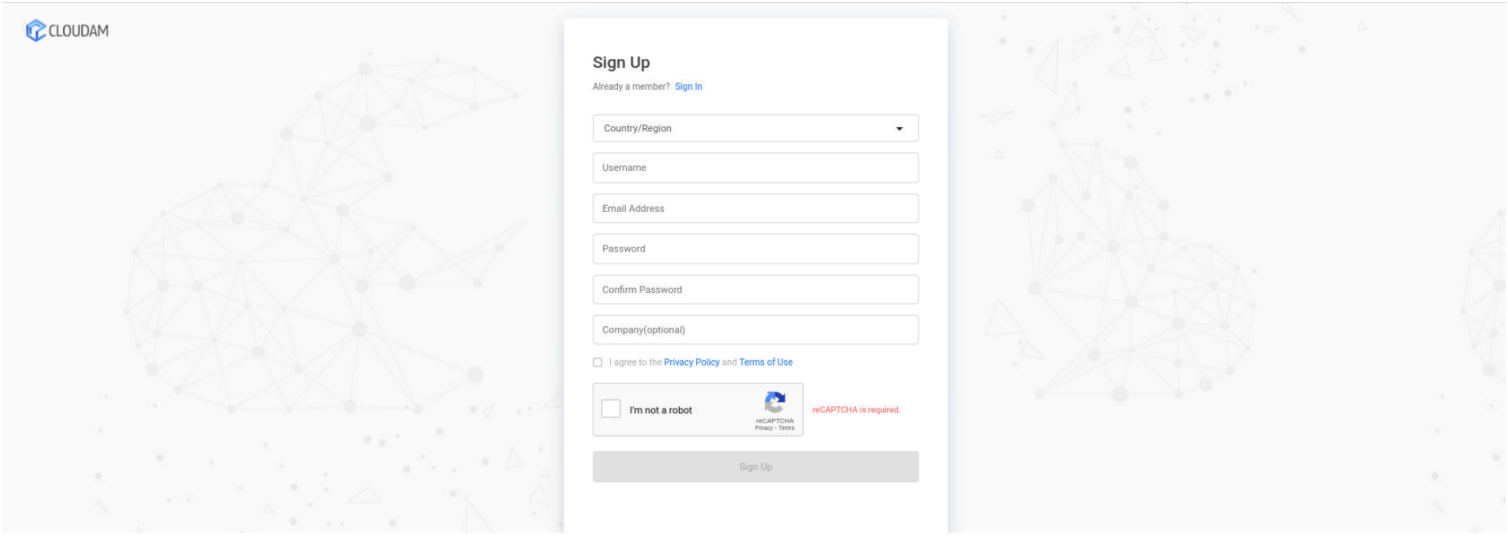
> Welcome to Cloudam Elastic HPC platform! This document provides some links to help you use this platform.

1. How to log in

Should you already have an account, please log in via https://www.cloudam.io/v2/pages/Internlogin-page



Don't have an account, please visit to register https://www.cloudam.io/v2/pages/Internregister-page


CLOUDAM
Sign Up
Already a member? Sign In
Country/Region
Username
Email Address
Password
Confirm Password
Company(optional)
I agree to the Privacy Policy and Terms of Use
I'm not a robot
reCAPTCHA
Privacy - Terms
reCAPTCHA is required.
Sign Up

Basic Concepts

1. Workspace

> An area based on the public cloud which has different hardware resources, Jobs, Data as well as HPC clusters are located in it. Different workspaces share nothing, user must manually copy data between them.

- **General Workspace**

  This computing area contains various CPU and GPU resources, which can meet the needs of most HPC jobs.

2. Terminal

> The E-HPC platform uses SLURM as the job scheduler, Terminal is the Linux SSH connection towards SLURM cluster, this function is suitable for people who are familiar with the HPC cluster and especially the SLURM job scheduler, user is able to submit HPC jobs to the SLURM cluster through the SSH terminal.

- **Master Node**

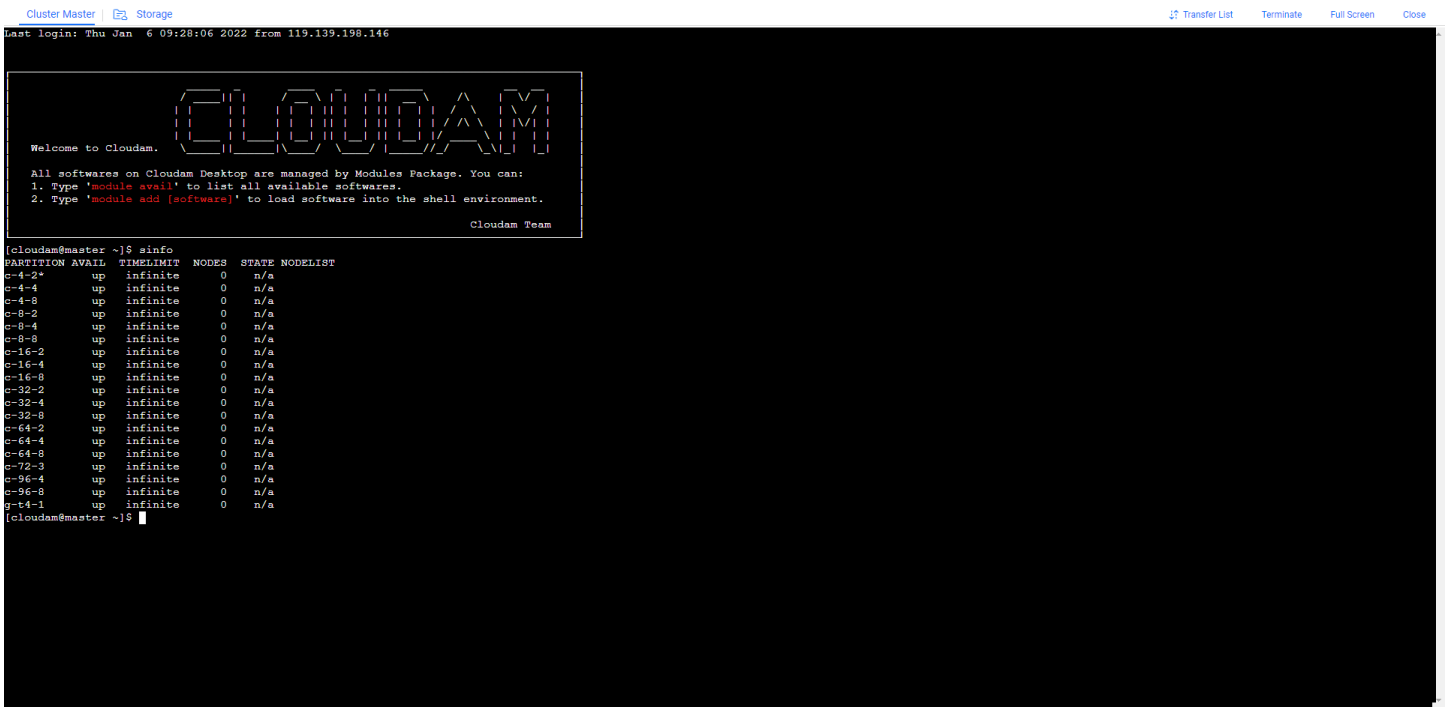
  The SLURM master node, after successful launched, it is then used for job submission and managements via SLURM CLI.

- **Worker Node**

  The Linux servers which are actually executed HPC job. Workers nodes are always launched after job submission request is recevied and automatically terminated once job is done(success/failed).

- **Job Partition**

  For node resources of different specifications, use the `sinfo` command to view the current partitions.



Partition example:

c-4-1		g-t4-1	
c	Indicates that the specification is a CPU node	g	Indicates that the specification is a GPU node
4	Indicates that the number of CPU cores on the node is 4 cores	t4	Indicates that the graphics card model is t4
1	Indicates that each core has 1G memory, that is, the node specification is 4 cores and 4G	1	Indicates that the number of GPU cards per node is one card

3. Desktop

> A remote desktop workstation has graphical interface, which is an independent Linux node rather than a node inside a SLURM cluster.

- **Linux Workstation**

The CentOS 7 operating system supports installing software and submitting HPC jobs on a single node via OpenMP.

- **VM Image**

  After installing the software on a luanched desktop, `Save as Image` can save the installed software and data to a customized image, and there is no need to reinstall the software when using it next time. The saved image can be shared with other users.

4. Submit Job

> The whole process is operated in the form of web console. Using the software template made by the platform, you only need to upload the input file and set the job parameters. The job results and operating logs can be viewed on the console page.

- **GUI Template**

  The visual template is a software submission script made in the background. When using it, you only need to upload the job input file and configure the running parameters to perform the job.

- **Demo Case**

  The demo case is a software submission script made in the background, it also provides a case file for demonstration. You only need to select the hardware configuration to submit the job directly. It is mainly used to test against the E-HPC platform.

5. Image Center

- **What is Image?**

  The image includes the operating system and preinstalled software. The image can only contain the basic operating system, or it can integrate a specific software environment on this basis.

- **The Role of the Image Center**

  The image center is convenient for you to install the software and create custom images. The customized workstation images can be viewed on the graphical interface.

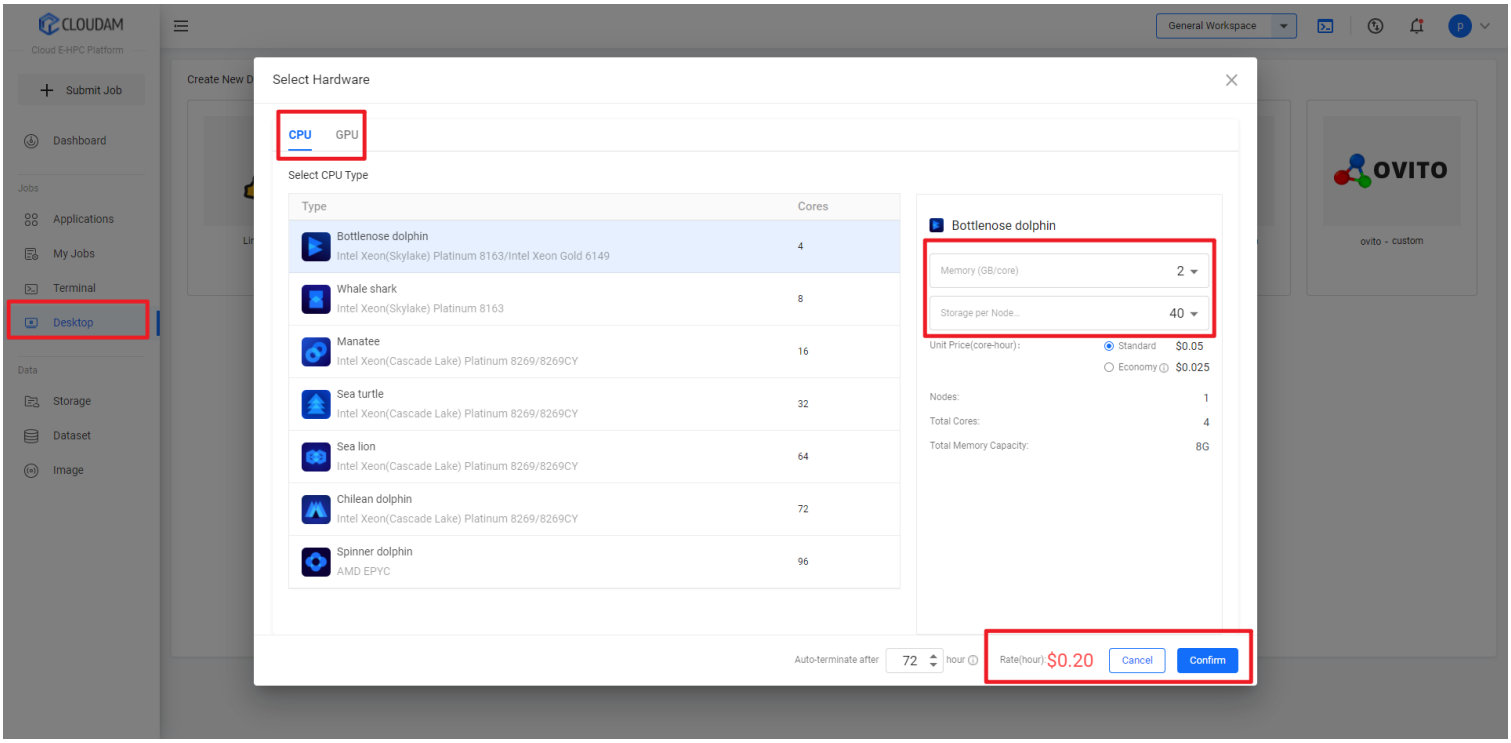
Pricing

1. Charge Items

Charge item	price
Compute	The price of resources differ, subject to the price listed when selecting the hardware configuration
File Storage	200GB free tier usage, above 200GB will be charged at rate of $12 per day per TB
Data disk	$0.15 per hour per TB

2. View the Real Time Pricing

A real time pricing information could be found from the Pricing menu of the Cloudam web console.



Quick Start

> This section briefly introduces the main functions of the E-HPC platform, so that you can get started quickly.

1. Dashboard

The Dashboard is the real-time data analysis dashboard provided by the Log Service, allowing you to keep track of accounts, messages, jobs, and resource usage in a timely manner.

2. Applications

You can search for the software you need here, click Submit after the application is approved, and select the method you are used to submitting a job.

3. My Jobs

The management interface for submitting jobs using templates allows you to view job status, connect to running computing nodes, and view historical jobs and other information.

4. Terminal

This function is suitable for users who have a certain understanding of SLURM Commands, submit jobs through the command line throughout the process, and support single/multi-node parallel computing.

5. Desktop

Workstations with graphical desktop operations can only perform parallel computing on a single node.

6. Storage

Provide upload and download of input files and result files, and support online viewing.

7. Dataset

Datasets can be understood as labeled data in machine learning scenarios, which are divided into public data sets and private data sets. Public data sets can be applied for by all enterprise users, while private data sets are only visible to users under the enterprise account.

8. Image

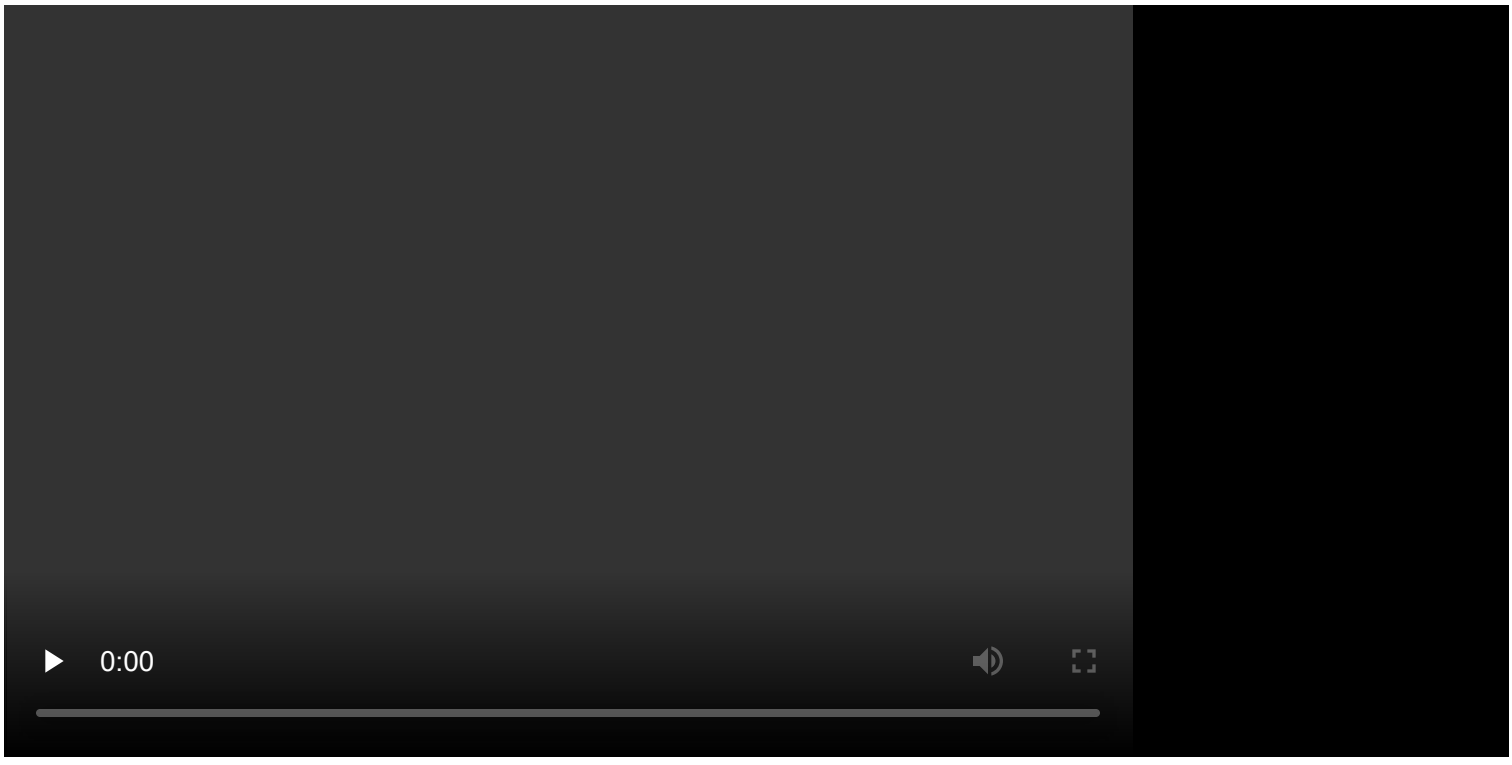
The image center is divided into virtual machine images and container images. Virtual machine images are convenient for you to install software and create custom images to be called during computing; in container images, you can create new repository and push docker images to run containerization applications.

Job Submission

> The Cloudam Elastic HPC platform has three job submission methods, which are divided into `template submission`, `CLI submission` and `desktop submission`. You can choose the job submission method according to your needs and usage habits .

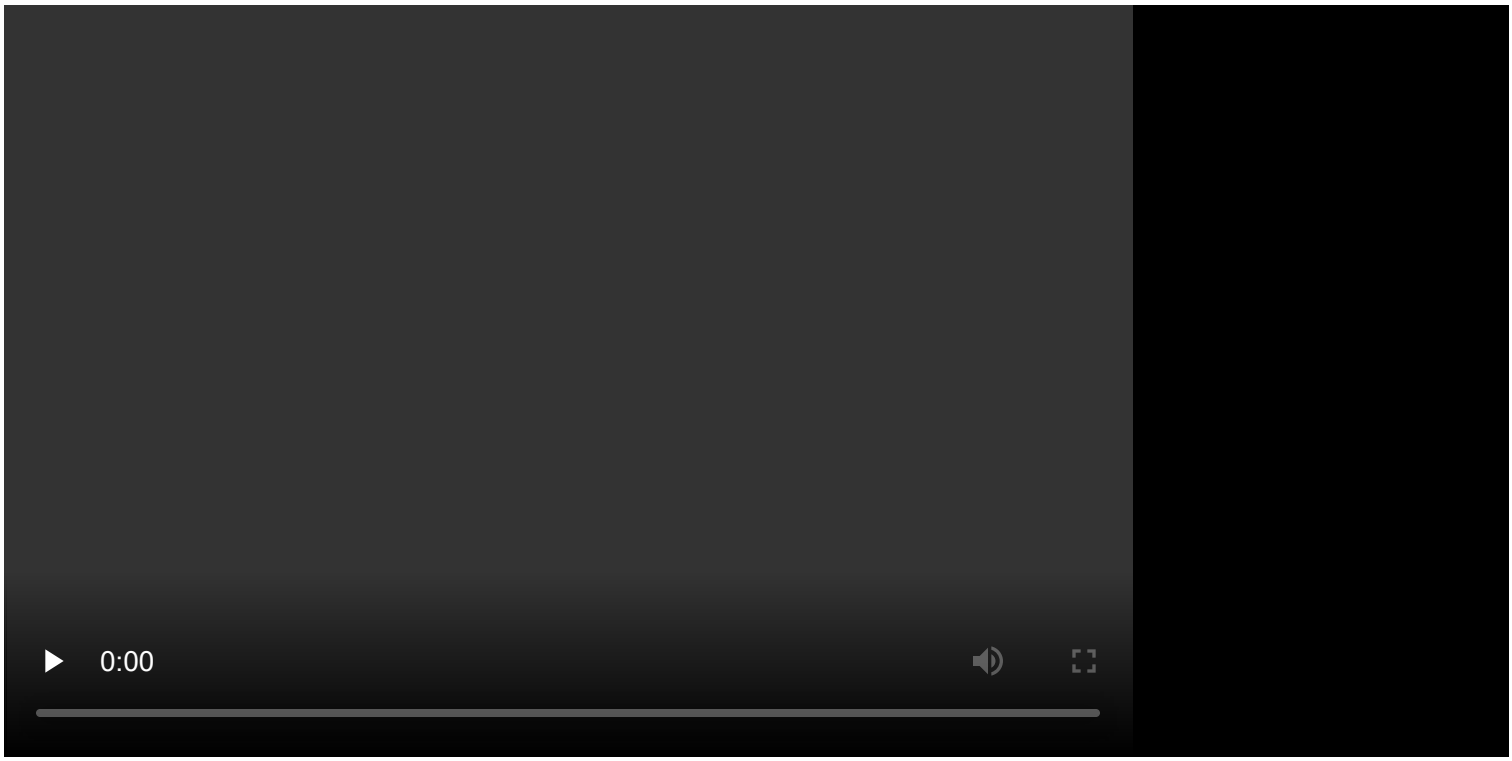
1. Template Submission

It is suitable for users without Linux commands experience. The job submission process and configuration parameters are fully visualized. Select the software you need in the application center interface. After the application is approved, select the template to submit, upload the input file, and then select the hardware configuration to submit the job.



2. CLI Submission

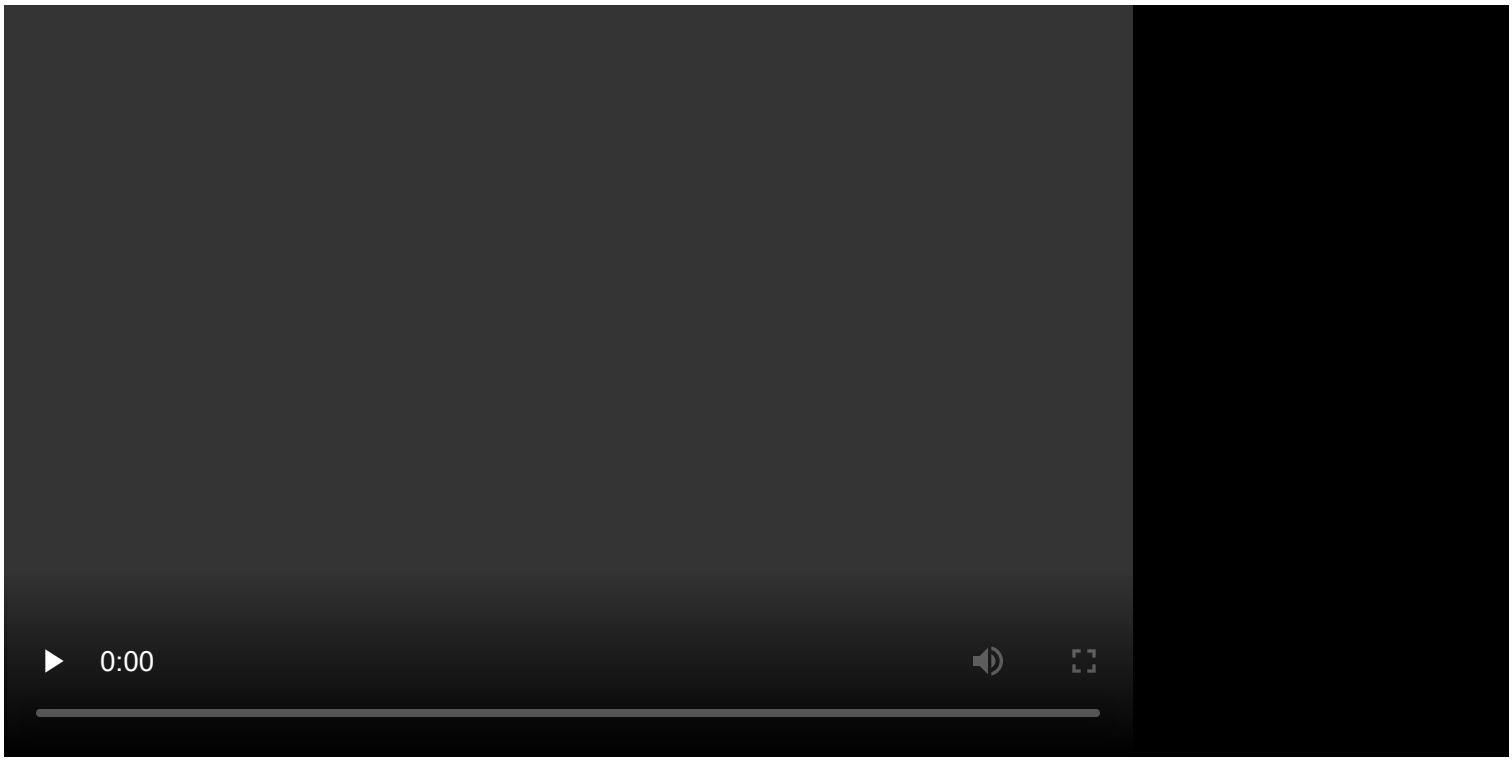
To create a new Terminal through the console, which is a SLURM cluster containing different hardware partitions, submit your job through the SLURM commands. The whole process only requires command line operations.



3. Desktop Submission

For softwares which have GUI interface, you are able to use it same as the way in your personal computer, select the software you need in the application center interface. After the application is approved, click Submit on the graphical interface, select the software version, start and connect to the workstation to use the software.

Note: desktop submission can only run single node job.


0:00

E-HPC Platform Editions

> In order to meet the differentiated needs of different users, E-HPC platform provides users with four editions: Standard, Enterprise, Dedicated, and On-premise. Users can contact the sales manager to activate according to your own needs.

1. Introduction

- **Standard:** The basics for individuals, support all needs as a HPC user.
- **Enterprise:** For enterprises, team organizations, scientific research institutions, etc., users who need to use unlimited computing resources, team management functions, and higher security levels.
- **Dedicated:** It is used for enterprise users with higher requirements, and supports all functions of the enterprise version, completely isolated environment for critical enterprise business.
- **On-premise:** It is used for organization who needs to manage on-premise HPC cluster as well as cloud bursting, it's a private deployment by which the organization has full control over.

2. Functionality Comparison

> Check out the link here to see the detailed differences between these editions, click here.

Account Management

> The account management includes function settings such as profile, billing&cost and payment etc. Log in to the Cloudam web console - click the personal icon avatar to enter the account management.

1. Profile

> The profile supports login password modification, API Token generation and other function settings.

2. Billing & Cost

> Users can manage E-HPC cost usage through this page, e.g. monthly cost, detailed cost records for a specific date, as well as looking into cost spent per job.

3. Payment

> Your payment information clould be viewed and modified via this page, e.g. update or remove an invalid credit card.

4. Pricing

billing rules

> The activation resource is accurate to the second, and the deduction is refreshed every hour. The billing time is within 1 hour after the end of the current billing cycle. The specific deduction details can be found in the Charge Center > Bill Details.

For example, the bill for 10:00-11:00 will be generated before 12:00, depending on the system billing time.

Notes: Before using, you need to ensure that there is no arrears in the account.

> The Pricing page can query the charging information of related resources.

5. Network Policy

> On the **Network Policy** page, you can enable custom network policies configured by administrative users or security administrators as needed. After enabling, it will only take effect on your own newly-launched nodes.

Related Tips

- For security reasons, network policies can only be modified and configured in the security management through the management account or security administrator. If you need to modify it, please contact the account administrator or security administrator to modify it.
- When an user enables a custom network policy configured by an administrative user or a security administrator, it only takes effect on the newly-launched nodes of the user.

Security

> This is an enterprise edition function, The Cloudam E-HPC platform has opened the security management function for enterprise users for the sake of security management. The settings in the security management are global security settings for the account, which will take effect for all users under the account.

Notes: The security management function is only open to the enterprise edition and above, and the standard edition does not support the security management function.

1. Basic Settings

> Basic settings can set the account login policy, shared data read/write permissions, data transmission encryption, node outgoing public network traffic control, etc. The settings will take effect globally for the account.

2. Key Pairs

> Account or users can connect to cluster nodes through SSH key pair, which is a safe and convenient way of login authentication. It consists of public key and private key. It currently only supports Linux nodes and SSH keys. For generating a pair of keys through an encryption algorithm, RSA 2048-bit encryption is used by default. To use an SSH key pair to log in to a Linux node, you must first create or upload a key pair, and then enable the SSH Authentication through the Basic Settings page.

Related Tips

- After successfully creating an SSH key pair：
  - Cloudam will save the public key part of the SSH key pair. In the Linux node, the public key content is placed in the ~/.ssh/authorized_keys file.
  - You need to download the private key and keep it securely, which is in unencrypted PEM (Privacy-Enhanced Mail) encoded `PKCS#8` format.
- Security: SSH keys are more secure and reliable for login authentication.
  - The security strength of the key pair is much higher than that of the regular user password, which can prevent the threat of brute force cracking.
  - It is not possible to derive the private key from the public key.

- Convenience：
  - It is convenient to remotely log in to the Linux node, which is convenient for management. If you need to maintain multiple Linux management nodes in batches, this method is recommended to log in.

Notices:

1. The platform does not save private key information, please keep your private key properly, and then use the private key to connect to the instance.
2. If an SSH key pair is used to log into a Linux instance, password login will be disabled for increased security.
3. Only the Linux node or desktop workstations started by the Linux system are supported, the Windows node does not take effect.
4. When the key pair is binded through the basic settings of the console, it is a global security setting for the account, which will take effect on the newly launched Linux nodes or the desktop nodes, the existing nodes will not be affected.

3. Network Policies

> In order to meet the customized needs for enterprise customers, Cloudam E-HPC platform adds new network policies settings. Network policies are a kind of virtual firewall with stateful packet filtering function. They are used to set network access control of cluster nodes. It is an important means of network security isolation.

Network Policy Rules

A network policy rule consists of the following components:

- Authorization object: IP or IP range of source data (inbound) or target data (outbound).
- Protocol Type and Protocol Port: Protocol type such as TCP, UDP, etc.

Notices

1. The network policy function is only available to the enterprise edition and above, and the standard edition does not support the network policy function.
2. The default rules of the network policy do not support operations such as changing, deactivating, deleting, and cloning. Except for the ports specified by the enabled default policy, the incoming and outgoing traffic of other ports are denied by default.

3. Network policies function can only be accessed by administrative accounts and security administrators. The enabled network policies and modified rules take effect on the fly, however it only affects master and desktop nodes, worker nodes are not allowed to change network policies.
4. Network policy names only support english letters, numbers, and special characters (. _ -). A maximum of 3 network policies can be created and enabled for each workspace, and a maximum of 25 rules can be created for each policy.
5. Each user can enable custom network policies configured by administrative users or security administrators in the Network Policy menu bar according to their own needs, and only take effect on the user's own newly-launched nodes.

Team Management

> To meet multi-department scenarios such as enterprises, schools, scientific research institutions, and team organizations, Cloudam E-HPC Platform has introduced a team management function.

1. Accounts

> An orgnization could have more than one account, each represents one team/department under Cloudam platform, an orgnization admin role account could create, edit, freeze and delete accounts on behalf of the orgnization. While each account with account admin role cloud create, edit and delete users under its account.

Account Roles

> Cloudam uses RBAC(Role Based Access Control) mechanism to control access for enterprise users, it currently supports several roles including Org Admin, Account Admin, Security Admin and System Admin.

- **Org Admin:** An orgnization has and only has one Org Admin role account at Cloudam.io, This role represents the orgnization itself in Cloudam E-HPC platform, which could be used to account creation as well as billing&fininal statstics.
- **Account Admin:** An account role represents a team of the orgnization who uses the E-HPC platform, e.g. an enterprise which has two R&D teams in Stockholm and Beijing respectively cloud have two accounts in Cloudam.io; billing and payment is currently at account level.
- **Security Admin:** Security related management function is open to this role.
- **System Admin:** System settings against the account and all its users is controled by this role.

2. Users

> An user is an E-HPC platform user under Cloudam, account admin could create users for team members of the team.

System Management

1. Quota Management

> In order to prevent resource abuse, the platform limits the resource quota of each computing area, and sets quota limits on the number and capacity of resources. The quota includes the number of CPU cores, the number of GPU cards, the number of images, the number of network policy creation, the number of sub The number of users created, etc., the quota is effective for all users of the platform. The resource usage of all sub-users under the main account shares the same main account quota, and only the main account or the system administrator can apply for it in the system management.

Quota Management

Quota Name	Description	Total Quotas	Scope
Number of GPU cards	The maximum number of GPU cards that current account can have, including those of GPU compute nodes and GPU desktop	10	Workspace
Number of CPU cores	The maximum number of CPU cores that current account can have, including those of CPU compute nodes and CPU desktop	400	Workspace
Number of customized images	The maximum number of custom images that can be owned by current account	5	Workspace
Number of customized network policies	The maximum number of custom network policies that can be owned by current account	3	Workspace
Number of users	The maximum number of users that can be owned by current account	100	Global

Quota Name	Description	Total Quotas	Scope
Number of accounts	The maximum number of users that can be owned by current organization	20	Global
Number of custom workspaces	The maximum number of custom workspaces that can be owned by current account	3	Global

Introduction to Storage Directory

> The E-HPC platform has a NAS with a storage capacity of up to 10PB in a single workspace.

Warning: The following directories cannot be deleted by themselves, otherwise the related functions cannot be used.

Directory	Description
/home/cloudam	The home directory of cloudam users, this directory is the shared storage of all nodes. For Linux software that does not require root permissions, it can be installed in this directory and saved permanently; for software that requires root permissions, it can be installed through the Image center and defined as a VM image
/public	The installation directory of the pre-installed software on the platform, you do not have permission to write in this directory

Data Transmission

> Data transmission allows users to upload job input files and download job result.

Notices:

- When uploading a file or folder, **the length of the name must not exceed 127 bytes, special symbols and spaces are not supported**;
- In the transfer list, click "Upload File" to upload it to the path of `/home/cloudam` by default. It cannot be uploaded to a custom directory. To upload to a custom directory, you need to enter the absolute path of the directory and click New to upload. The uploaded custom folder must have The cloudam user has write permission and cannot be created by the root user, otherwise uploading cannot be performed.
- Refreshing the web page during file transfer will cause the transfer to fail.

1. Download the Result Files

> Jobs submitted by the visualization template, successful or failed jobs can be viewed in the job management on the left menu bar to view the result file, log file, input file, temporary file, download the corresponding result file or log file, click the "Download" button, the system will automatically package and compress and download to the local computer.

Notices:

- Click "Download", the waiting time is long, maybe because the result file is large, or the downloaded folder is relatively large, the system automatically compresses and packs it, please do not click the "Download" button multiple times.
- Each workspace has its own storage server, and file does not share between workspaces. Please select the working directory of the corresponding workspace to download the corresponding result file.

Steps:

1. Select Workspace -> My Jobs, and click View;

Select the file or folder to download and click Download.

Search Platform Pre-installed Software

> The E-HPC platform integrates a variety of application softwares, such software can be directly loaded without repeated installation.

1. Use CLI to Query Software

> **Module** is an environment variable management tool. Cloudam E-HPC platform has installed a lot of public softwares, and through module manage environment variables.

Step 1. Start and connect to a Terminal;

Step 2. List all pre-installed softwares on the platform;

```
module avail
```

Step 3. Query the specified software (such as query lammps);

```
module spider lammps
```

2. Query for Python/Conda Environment

> **Anaconda** is an open source package management system and environment management system for installing multiple versions of packages and their dependencies, and easily switching between them.

Step 1. Load Anaconda3

```
module add Anaconda3
```

Step 2. List the platform's preinstalled Conda virtual environments

```
conda env list
```

Step 3. Execute `source activate` to enter the virtual environment with pre-installed software

```
source activate xxxx
```

3. Query Desktop Softwares

> Search the software name in the application center. You might need to apply before you could use it. For the software that has approved, you can select the version installed on the platform. Select **Desktop** to start a desktop workstation.

4. Do not Find the Required Software

1. You can install and configure custom software or environments through **Image**. If you have any questions during the installation process, you can contact customer service for help.
2. We evaluate common open source software for global deployment and welcome feedback.
3. The platform has integrated most common free and open source softwares for scientific research application scenarios. For commercial software, you can purchase licenses and install them by yourself.

Load Preinstalled Software

> All pre-installed software on the platform can be loaded directly through the command line.

1. Use Linux Commands

1. Use the module tool to query and load software

```
module avail            #List all preinstalled software for the platform
module spider xxxx      #Query the specified software
module add xxxx/xxx     #Load the specified version of the software
```

2. Load the Python/Conda virtual environment

```
module add Anaconda3    #Load Anaconda3
conda env list          #List all Conda virtual environments
source activate xxxx    #Load the specified virtual environment
```

2. Desktop Software

Step 1. Enter the application center to search for the software name, check the installed version of the software, and select `desktop submission`;

Step 2. Select the hardware configuration and start;

- CPU or GPU: According to the installed software version and usage experience, choose whether the computing resource is CPU or GPU;
- Memory ratio: set the workstation memory size as the number of cores x memory ratio;

- On Linux workstations, you can use the right mouse button on the desktop of a single computer, open the terminal view, and enter the software installation directory to start.

DIY Software Installation

1. Installation Instructions

Choose right way to install software

- **Installation via terminal**

  This is the default and suggested way to install software, apply for worker node and then install software environment to the `/home/cloudam/Software` directory, which can be used by the SLURM cluster.

- **Installation via Python/Conda environment**

  The environment created by Conda and the installed software are saved in the `/home/cloudam/.conda` directory by default, which can be used by both cluster nodes and Linux workstations.

- **Installation via desktop**

  It is suitable for softwares which has GUI interface and uses as only in single node environment. The software installed in the directory other than /home/cloudam will not be persistant once the desktop node is terminated.

- **Installation via image center**

  In case the software installation procedure `requires root privilege`, such as "sudo", "yum" or "apt-get" commands etc. Install it through image center step by step is suggested. Software installed with root privileges on all cluster nodes or Linux workstation nodes will not be saved in the /home/cloudam directory by default, thus save the installation as a VM image is needed, SLURM cluster started after that will use the VM image you saved to start the corresponding SLURM nodes.

Notices before installation

- For software programs that require root privileges to install or use, it is recommended to install and create a custom image through the image center to persist the installed software.

- The software installed directly on the cluster ssh management node may not work properly, because the software always runs on the computing node. To ensure a consistent operating environment, the installation needs to be performed on the computing node.
- HPC software usually relies on third-party libraries such as mpi, fftw, etc. These third-party libraries have basically all been installed on the E-HPC platform, just use `module avail` or `module spider xxxx` to query, and then activate it through `module add xxxx` to use it. For more software loading of third-party dependent libraries, please click to view Load Preinstalled Software.

2. SSH and CLI software installation

Installation via terminal

Step 1. Click the Terminal to start SSH head node, then ssh connection to a terminal head node;

Step 2. Execute the command `salloc -N 1 &` to start a worker node and connect to ssh;

Step 3. View the dependent libraries installed on the platform, and then load the software dependent libraries. For more software loading of third-party dependent libraries, please click to view Load Preinstalled Software;

```
module avail          #view all
module spider xxxx    #quick search
module add  xxxx      #load
```

Step 4. Upload the installation package via storage;

Step 5. Start compiling and installing source code programs, e.g.;

You need to specify the installation path to install it in the /home/cloudam subdirectory. Such as `./configure --prefix=/home/cloudam/Software`

```
./configure   #Used to generate Makefile
make          #Start source code compilation
make install  #start installation
```

Step 6. After the installation is complete, execute `scancel JOBID` to release the worker node and complete the installation; Step 7. After the installation is successful, an executable program will be generated in the directory;

Python/Conda environment installation software

```
#Load the Anaconda environment
module add Anaconda3/2020.02

#Create an environment, the default installation path is /home/cloudam/.conda
conda create -n xxxx

#View the created environment
conda env list

#Load the created environment
source activate xxxx

#Start installing the software in the created environment
conda install xxx xxx
```

For more common commands of Conda, please refer to Conda usage

3. Installation via desktop

Step 1. Start a Linux desktop workstation and connect to it;

```
sudo systemctl restart vncserver@:1.service
```

2. Linux Graphics Workstation **iOS** Image Installation Software

```
mkdir -p /tmp/mount                          #Create a mount directory under the /tmp
path.
chmod -R 755 XXXXXXX.ios                     #Add permissions to the IOS image file.
mount -o loop XXXXXXX.ios /tmp/mount         #Mount the image file to the specified
directory.
export DISPLAY=:1                            #Defines the graphical desktop environment
variables.
cd /tmp/mount                                #Enter the mount directory to execute the
installation program. The installation path needs to specify the /home/cloudam path.
Subsequent graphics workstation releases can be permanently saved, and you can use it
again to enter the specified directory to start.
```

Step 1. Start a Linux workstation and connect to it;

Step 2. To view the dependent libraries installed on the platform, load the dependent libraries required by the software, and to load more software of third-party dependent libraries, please click to view Load Preinstalled Software;

```
module avail          #View the existing software environment of the cluster
module spider xxxx    #Find software environment
module add  xxxx      #Load software environment
```

Step 3. Upload the installation package via storage;

Step 4. Start compiling and installing the source code program;

You need to specify the installation path to install it in the /home/cloudam subdirectory. Such as `./configure --prefix=/home/cloudam/Software`

```
./configure   #Generate Makefile
make          #Start source code compilation
make install  #start installation
```

Step 5. After the installation is successful, an executable program will be generated in the directory; Step 6. After the installation is complete, terminate the Linux workstation node;

4. Installation via image center

> The image center is convenient for you to install software and make custom image to be used during job running.

- A custom cluster image can be set as the `default cluster master/compute image`, and subsequent cluster jobs you submit will use this image to create corresponding nodes.
- The customized `desktop image` will be displayed in the launch options of the desktop application, and you can edit and submit jobs directly on the launched desktop.
- When root privileges are required, enter the following command to gain temporary root privileges: `sudo -i`.
- The operating system of this platform is CentOS 7, please confirm whether the installed software version matches the operating system.

How to submit a job?

> Cloudam E-HPC platform has multiple ways to submit jobs for different usage scenarios of users. How to choose?

Usage requirements	Suggested submission method
Unfamiliar with Linux commands, unfamiliar with software parameters	Template submission
Need to use multiple nodes to run jobs in parallel, but not familiar with SLURM commands	Template submission
I have used the HPC cluster before, and I am familiar with the SLURM commands	command line submission
Only need to use a single Linux node to run jobs	Desktop submission (Linux workstation)

Template submission

> Search the software in the application center, select the software, click Submit Job, select the template to submit, upload the job input file, set the template parameters, select the configuration, and you can view the results in real time after submitting.

1. Submission

Step 1. Select the software in the application center, you can view all the submission methods of the software;

Step 2. Select `Template`;

- Due to some software limitations, if there is only a basic template, the software `does not support using template submission`.
- The demo case template is an example of the input file pre-uploaded in the background. It is for reference only and does not support modification.

Step 3. Upload the input file and configure the job parameters;

- The submitted job name can be modified.
- You can `click the info mark` to view the format of the input file to be uploaded.

Step 4. select hardware configuration;

- Select the CPU/GPU configuration according to the software version and usage requirements.
- Number of nodes: Set how many computing nodes to start parallel computing.
- Memory ratio: Set the memory size of each computing node as the number of cores per node × memory ratio.

Step 5. View the summary of the job and then click submit button;

2. Monitoring after submission

1. After the job is submitted, you can view the output and log of the job through `My Jobs`.
2. You can log in to each computing node through an ssh connection to view the detailed information of the node.
3. You can view information such as resource usage of running computing nodes through the `Dashboard`.
4. You can configure notification such as email notification after job completion in `Notification Settings`.

Command Line Submission

> The command line submission uses the SLURM command, and submits jobs to one or more computing nodes in the form of scripts for parallel computing, which can be parallelized by MPI or OpenMP. If you have used the corresponding script on other HPC platforms, you need to provide it to the technical engineer to modify it into a script suitable for the E-HPC platform.

1. Submission Steps

Step 1. Start a head node through the Terminal, and connect to the head node;

Step 2. Use storage to upload job input files, regarding how to upload files, please click to view Data Transmission;

Step 3. View the partitions supported by the SLURM cluster;

```
sinfo
```

Step 4. To query and load software, please click to viewd Load Preinstalled Software;

```
module avail         #view all
module spider xxxx   #quick search
module load  xxxx    #load
```

Step 5. Create a submission script, below is a GROMACS job script: `su.sh`

```
#!/bin/bash
module add GROMACS/2021-gromacs-cpu-new
mpiexec -v  gmx_mpi mdrun -v  -cpi tpr_file_name -deffnm tpr_file_name
```

Step 6. Use sbatch to submit to the worker node. For details of parameters, please refer to the SLURM Commands;

```
sbatch -N 2 -p c-4-1 -n 8 -c 1 su.sh
```

Step 7. View the job's status in the SLURM cluster;

```
squeue
```

Step 8. Connect to the worker node and execute `top` command to view the CPU usage;

```
ssh c-4-1-worker0001    #Connect to compute nodes
top                     #View Task Manager
exit                    #Exit the compute node
```

Step 9. View job running details;

```
scontrol show jobs
```

- /home/cloudam/examples/GROMACS is the job execution path.
- The /home/cloudam/examples/GROMACS/slurm-47.out file is the job output log, and you can view the job running information in real time.

Step 10. Cancel the running of the program and release the allocated computing nodes (the computing nodes will be automatically released if the job execution succeeds or fails);

```
scancel JOBID
```

Step 11. How to download the result file, please click to view Data Transmission;

2. Worker node resource utilization monitoring

1. View resource usage such as CPU and memory through the `Dashboard`.
2. Use the `Monitor` button on the terminal node to view the resource usage. You can log in to the compute node and use commands to view detailed information.

Desktop Submission

> Provides a graphical user interface (GUI), supports Linux operating systems, and can remotely connect to workstations with protocols such as VNC, SSH, and so on.

1. Linux Desktop Workstation Submission

Step 1. Search for the software you need in the Desktop page. After the application is approved, click Submit on the graphical interface, select the software version, and connect to the workstation to use the software. If you do not find the software you need, you can also install the software by yourself. For details, please see Linux workstation installation software;

Step 2. Select the hardware configuration, start;

Step 3. Click on the VNC connection, right-click to open the terminal, and enter the startup command to use the software;

Step 4. Pay attention to various functions of a desktop;

- Terminate: When the workstation is not needed, the workstation can be released, and the running application will be shutted down of course.
- Save as Image: The software installed outside the /home/cloudam directory needs to be persisted by saving as image.
- Settings: Functions such as `Disable Hyper-threading` and `Notify Job End` can be configured.
- Insight: View the node resource usage for current or historical jobs.

AlphaFold2

> AlphaFold is a protein structure prediction tool developed by DeepMind (Google). It uses a novel machine learning approach to predict 3D protein structures from primary sequences alone.

`Notice`

- The input file name cannot contain special characters or Spaces,a file ending in '.fasta 'format.
- The submitted file is first subject and second subject sequence，The sequence must be uppercase and must not contain spaces or newlines.
- If a job contains multiple sequence files at once, multiple compute nodes will be launched, each compute node will handle one sequence file.

Monomer input file example：

```
>sequence_1
GSDNGFGSSKATSGSDFGGLAIFDGSGSEHFGHSDTHGSFDGLFGVDFZILSQQLKS
```

Multimer input file example：

```
>sequence_1
FADGAFGSGSDFSGFHGFGGHSRADGGTGFDGDFSNGLFNGFGVLTFSDERGESGFDGFFDSGSFQGKFDOFJOWFJODJOAISJFDO
>sequence_2
GSDNKDSKATSEREACGLAIFSKQHGFSHSDFGSFFILQFDSAFQLKSKFDOASFJAPOFJDPAISFJPAFJKAPF
```

1. Template Submission

> Mulple input files could be provided at once, for example: upload five input files and set the number of nodes to 1, then submit the job, five worker nodes are launched to compute simultaneously rather than one node to do the prediction sequentially.

Step 1. Search for AlphaFold software in the Application center, select Submit; Step 2. Select GUI template; Step 3. Uploading sequence files(fasta format) ; Step 4. Select GPU hardware configuration; Step 5. View the job summary and submit the job;

2. Command Line Submission

Use SSH to connect to a terminal head node

Step 1. Create the job directory and enter;

```
mkdir alphafoldJob1
cd alphafoldJob1
```

Step 2. The input file required for uploading is test.fasta. For details, see Data Transmission. Step 3. Create the following execution script in this folder `alphafold.sh`:

```
#!/bin/bash
#Grant read and write permission to the output directory
chmod -R 777 /home/cloudam/alphafoldJob1

#Initialize the program running environment
module add Anaconda3
PROGRAM=/public/software/.local/easybuild/software/alphafold
cd $PROGRAM/alphafold-2.1.1
docker load < alphafold2.1.tar
pip install -r docker/requirements.txt

#If be multimer
python docker/run_docker.py \
--fasta_paths=/home/cloudam/alphafoldJob1/test.fasta \
--is_prokaryote_list=false \
--max_template_date=2021-11-01 \
--model_preset=multimer \
--data_dir=$PROGRAM/data2 \
--output_dir=/home/cloudam/alphafoldJob1
```

Step 4. Submit Job;

> Submit job to run on a GPU node with a V100 card.

```
sbatch -p g-v100-1 -c 8 alphafold.sh
```

For details about job running status and parameters, click The SLURM command.

3. Introduction to the Result Files

A normal alphafold result should contain the following file structure:

```
<target_name>/
    features.pkl
    ranked_{0,1,2,3,4}.pdb
    ranking_debug.json
    relaxed_model_{1,2,3,4,5}.pdb
    result_model_{1,2,3,4,5}.pkl
    timings.json
    unrelaxed_model_{1,2,3,4,5}.pdb
    msas/
        bfd_uniclust_hits.a3m
        mgnify_hits.sto
        uniref90_hits.sto
```

Ranked_0 to 4 are the five models with the highest scores predicted by AlphaFold2, ranked_0 is the best, please check below links for more details:

- AlphaFold GitHub:https://github.com/deepmind/alphafold
- AlphaFold Nature:https://www.nature.com/articles/s41586-021-03819-2

AutoDock-Vina

> AutoDock is a suite of automated docking tools. It is designed to predict how small molecules, such as substrates or drug candidates, bind to a receptor of known 3D structure.

`Notice`

- Input protein file in '.pdb' format.
- Other files (ligand libraries) require tar packages made from multiple sdf files, an sdf file can contain only one molecule.

- The default value for exhaustiveness is 3, you can change it by yourself.

1. Template to submit

Step 1. Search for Autodock & AutoDock Vina software in the Application center,please contact customer service for approval after application;

Step 2. Select GUI template;

Step 3. Upload target files,Ligand molecular library file;

Step 4. Select hardware configuration;

Step 5. View running jobs through My Jobs;

2. AutoDockTools

> AutoDockTools is a graphical user interface software developed to make it easier to use AutoDock. Using AutoDockTools, you can use AutoDock quickly without having to input Linux commands.

Step 1. Search for 'AutoDock Vina' software in the Application center; Step 2. Click `Submit`, select `Desktop` and then click launch button; Step 3. Start AutoDockTools, select the hardware configuration and start; Step 4. After a desktop workstation is launched, use VNC to connect to it.

CP2K

> CP2K is a program to perform atomistic and molecular simulations of solid state, liquid, molecular, and biological systems. It provides a general framework for different methods such as e.g., density functional theory (DFT) using a mixed Gaussian and plane waves approach (GPW) and classical pair and many-body potentials.

1. Command Line Submission

Use SSH to connect to a terminal head node.

Step 1. Create the job directory and enter into it;

```
mkdir cp2kJob1
cd cp2kJob1
```

Step 2. To upload the required input files via file transfer, see Data Transmission for details; Step 3. Create the following execution script in this folder `cp2k.sh`:

```
#!/bin/bash
module add CP2K/cp2k-7.1.0
mpirun cp2k.popt -i Ac.inp -o test.out
```

Step 4. Submit Job;

> Two 4-core nodes start eight parallel tasks.

```
sbatch -N 2 -p c-4-1 -n 8 -c 1 cp2k.sh
```

GROMACS

> GROMACS is a versatile package to perform molecular dynamics and energy minimization, i.e. simulate the Newtonian equations of motion for systems with hundreds to millions of particles. It is primarily designed for biochemical molecules like proteins, lipids and nucleic acids that have a lot of complicated bonded interactions, but since GROMACS is extremely fast at calculating the nonbonded interactions (that usually dominate simulations) many groups are also using it for research on non-biological systems, e.g. polymers.

`Notice`

- The tpr_file_name in the script command must be replaced with the TPR file name corresponding to the user.

1. Template Submission

Step 1. Search for Gromacs software in the Application center;

Step 2. Select GUI template;

Step 3. Uploading input files, it must contain only one tpr file, this software versions have CPU and GPU editions, select the corresponding version;

Step 4. Select hardware configuration;

Step 5. View the job summary and submit the job;

Step 6. View running jobs through My Jobs;

2. Command Line Submission

Step 1. Use Terminal to connect to a terminal head node; Step 2. Create the job directory and enter;

```
mkdir gromacsJob1
cd gromacsJob1
```

Step 3. Upload the files you need to run Gromacs to this folder，See Data Transmission for details.

GROMACS job example for GPU

Step 1. Create the following execution script in this folder `gromacs.sh`:

```
#!/bin/bash
module add GROMACS/2021-fosscuda-2019b   #加载软件
export GMX_GPU_DD_COMMS=true
export GMX_GPU_PME_PP_COMMS=true
export GMX_FORCE_UPDATE_DEFAULT_GPU=true

#Generate the tpr input file. If the tpr file already exists, no need to write
#gmx grompp -f pme.mdp -c conf.gro -p tpr_file.top -o tpr_file_name.tpr

gmx mdrun -v  -pin on -nb gpu -bonded gpu -pme gpu -cpi tpr_file_name -deffnm
tpr_file_name
```

Step 2. Use the SLURM Commands to submit to the compute node;

```
sbatch -N 1 -p g-v100-1 -c 12 gromacs.sh
```

GROMACS job example for CPU:

Step 1. Creating an execution script::

```

```

```
#!/bin/bash
module add GROMACS/2021-gromacs-cpu-new
mpiexec -v  gmx_mpi mdrun -v  -cpi tpr_file_name -deffnm tpr_file_name
```

Step 2. Submit job;

> Two 4-core nodes start eight parallel tasks.

```
sbatch -N 2 -p c-4-1 -n 8 -c 1 gromacs.sh
```

Jupyter Notebook

> The essence of Jupyter Notebook is a web application that facilitates the creation and sharing of literary program documentation, supporting live code, mathematical equations, visualization and markdown. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, machine learning, and more.

1. Desktop Submission

Step 1. Search for Jupyter Notebook software in the application center and select the Desktop to submit;

Step 2. Select the Desktop to launch;

Step 3. Select the hardware configuration;

Step 4. Connect to the activated workstation;

Step 5. Use the software;

LAMMPS

> LAMMPS is the Large Scale Classical Molecular Dynamics code, which stands for Large Scale Atomic/Molecular Massively Parallel Simulator. LAMMPS have potential for use in soft materials (biomolecules, polymers), solid-state materials (metals, semiconductors), and coarse-grained or mesoscopic systems. It can be used to model atoms, or more generally, it can be used as a parallel particle simulator at atomic, meso, or continuum scales.

1. Template Submission

Step 1. Search for the lammps software in the application center;

Step 2. Select a GUI template;

Step 3. Click the input file list to upload the file, which must contain at least one .in file;

Step 4. Select the hardware configuration;

- Number of nodes: Set how many computing nodes to start parallel computing.
- Memory ratio: Set the memory size of each computing node as the number of cores per node × memory ratio.

Step 5. View the job summary and then submit the job;

Step 6. View running jobs through My Jobs;

2. Command Line submission

Connect via SSH create and connect to a terminal head node.

Step 1. Create a job directory and enter;

```
mkdir lammpsJob1
cd lammpsJob1
```

Step 2. Upload the required input files. For details, please refer to Data Transmission;

Step 3. Create the following execution script lammps.sh in this folder:

```
#!/bin/bash
module add LAMMPS/3Mar2020-foss-2020a-Python-3.8.2-kokkos
ulimit -s unlimited
ulimit -l unlimited
mpirun lmp -in M-1.in
```

How to load more software versions, please click to view Load Preinstalled Software

Step 4. Submit the job;

> 2 4-core nodes, start 8 parallel tasks.

```
sbatch -N 2 -p c-4-1 -n 8 -c 1 lammps.sh
```

To view the job running status and parameter details, please click to view the SLURM Commands.

Please check Data Transmission to download the result file.

Click to download the above job example: LAMMPS.zip.

ORCA

> ORCA is a computational chemistry program focused on quantum chemistry applications designed to simulate the electronic structure as well as the spectral properties of molecules.

1. Command Line Submission

Step 1. Create a job directory and enter;

```
mkdir ORCAJob1
cd ORCAJob1
```

Step 2. Upload the relevant files needed to run ORCA to this folder. For details, please refer to Data Transmission; Step 3. Create the following execution script orca.sh in this folder:

```
#!/bin/bash
module add ORCA/5.0-openmpi-4.1.1
ulimit -s unlimited
ulimit -l unlimited
/public/software/.local/easybuild/software/ORCA/ORCA-5.0/orca_5_0_0_linux_x86-
64_shared_openmpi411/orca test.inp > orca.out
```

Step 4. Submit the job using sbatch:

> 1 4-core node, launch 4 parallel tasks.

```
sbatch -N 1 -p c-4-1 -n 4 -c 1 orca.sh
```

To view the job status and parameters, please click to view Slurm Job Management System.

Please check Data Transmission to download the result file.

Click to download the above job example: ORCA.zip.

PyTorch

> PyTorch is a Python-first deep learning framework and a GPU- and CPU-optimized deep learning tensor library capable of implementing tensors and dynamic neural networks on top of powerful GPU acceleration.

1. Command line submission

Connect via SSH create and connect to a management node.

Step 1. Create a job directory and enter;

```
mkdir pytorchJob1
cd pytorchJob1
```

Step 2. Upload the input file test.py through file transfer. For details, please refer to Data Transmission;

Step 3. Create the following execution script pytorch.sh in this folder:

```
#!/bin/bash
module add Anaconda3/2020.02 #Load Anaconda3 software
source activate pytorch-1.5 #Activate the pytorch environment
python test.py > py.log #Run the program
```

Step 4. Submit the job using the sbatch command;

> Submit the task to run on a GPU node with a T4 card.

```
sbatch -p g-t4-1 -c 4 pytorch.sh
```

Please click to view the SLURM Commands to view the detailed introduction of the job operation and parameters. Please check Data Transmissionto download the result file.

TensorFlow

> TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

1.Command line submission

Create and connect a terminal management node via an Terminal connection.

Step 1. Create a job directory and enter;

```
mkdir tensorflowJob1
cd tensorflowJob1
```

Step 2. Upload the input file tf.py. For details, please refer to Data Transmission;

Step 3. Create the following execution script TensorFlow.sh in this folder:

```
#!/bin/bash
module add Anaconda3/2020.02          #Load Anaconda
source activate tensorflow-gpu-2.2    #Activate the tensorflow environment
python tf.py > tf.log                 #Run the program
```

Step 4. Submit the job using the sbatch command;

> Submit a task to run on a GPU node with a T4 card.

```
sbatch -p g-t4-1 -c 4 tensorflow.sh
```

To view the job running status and parameter details, please click to view the SLURM Commands. For the result file download, please see Data Transmission.

TeraChem

> TeraChem's speed forms the basis for a suite of quantum chemistry applications, including optimization and dynamics of proteins, automated and interactive chemical discovery tools, and large-scale nonadiabatic dynamics simulations.

1.Command Line Submission

Create and connect to a management node via an Terminal connection.

Step 1. Create a job directory and enter;

```
mkdir terachemJob1
cd terachemJob1
```

Step 2. Upload the input files start.sp and complex.xyz. For details, please refer to Data Transmission;

Step 3. Create the following execution script terachem.sh in this folder:

```
#!/bin/bash
source
/public/software/.local/easybuild/software/terachem/terachem1.9/TeraChem/SetTCVars.sh
/public/software/.local/easybuild/software/terachem/terachem1.9/TeraChem/bin/terachem
start.sp
```

Step 4. Submit the job using the sbatch command;

Submit the task to run on a GPU node with a T4 card.

```
sbatch -p g-t4-1 -N 1 -c 4 terachem.sh
```

To view the job running status and parameter details, please click to view the SLURM Commands. For the result file download, please see Data Transmission.

VirtualFlow

> VirtualFlow is a versatile, parallel workflow platform for carrying out virtual screening related tasks on Linux-based computer clusters of any type and size which are managed by a batchsystem (such as SLURM).

Notices

- The input file is a protein file in .pdbqt format.
- For Docking parameter settings, please refer to the documentation of the specific docking program.
- VirtualFlow additional operation control parameter settings, refer to all.ctrl.

1.Template Submission

Step 1. Search for VirtualFlow software in the application center;

Step 2. Select the GUI template to submit;

Step 3. Upload the target file (protein file for Docking calculation, pdbqt format);

Step 4. Select the hardware configuration;

- Number of nodes: Set how many computing nodes to start parallel computing.
- Memory ratio: Set the memory size of each computing node as the number of cores per node × memory ratio

Step 5. View the job summary and submit the job;

Step 6. View running jobs through My Jobs;

2.Post-Processing Failure

> The job could run into fail sometime due to the post-processing error occurs, which can be fixed by the following steps.

Step 1. View the working directory running on the console, which can be viewed through output files, log files, etc. (eg: /home/cloudam/jobs/vf-demo/vf-demo_1611973330533); Step 2. Create and connect the management node via SSH connection; Step 3. Manually execute post-processing commands;

```
cd /home/cloudam/jobs/vf-demo/vf-demo_1611973330533
chmod 755 .postProcess.sh
rm -rf out/*
./.postProcess.sh
```

SLURM Commands

> Slurm (Simple Linux Utility for Resource Management, http://slurm.schedmd.com/) is an open source, fault-tolerant and highly scalable resource management and job scheduling system for large and small Linux clusters. HPC systems can utilize Slurm for resource and job management to avoid mutual interference and improve operational efficiency. All jobs to be run, whether for program debugging or business computing, must be submitted through commands such as interactive parallel srun, batch sbatch, or distributed salloc. After submission, you can use related commands to query the job status, etc.

1. Common commands

```
sinfo      #View partition status
squeue     #View jobs in the queue
scontrol   #View job details
scancel    #Cancel submitted jobs
sbatch     #batch submit job
salloc     #allocated running job
```

(1) View partition status

```
sinfo
```

- The naming rule of CPU partition is `c-number of cores-memory size per core`, such as c-8-4: indicates that the single node specification is 8 cores, and each core has 4G memory, that is, the node specification is 8 cores and 32G.
- The naming rule of GPU partition is `g-card number-number of cards per node`, such as g-v100-2: it means that there are two GPU nodes with graphics card model tesla v100.

```
[cloudam@master ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
c-4-2*       up   infinite      0    n/a
c-4-4        up   infinite      0    n/a
c-4-8        up   infinite      0    n/a
c-8-2        up   infinite      0    n/a
c-8-4        up   infinite      0    n/a
c-8-8        up   infinite      0    n/a
c-16-2       up   infinite      0    n/a
c-16-4       up   infinite      0    n/a
c-16-8       up   infinite      0    n/a
c-32-2       up   infinite      0    n/a
c-32-4       up   infinite      0    n/a
c-32-8       up   infinite      0    n/a
c-64-2       up   infinite      0    n/a
c-64-4       up   infinite      0    n/a
c-64-8       up   infinite      0    n/a
c-72-3       up   infinite      0    n/a
c-96-4       up   infinite      0    n/a
c-96-8       up   infinite      0    n/a
g-t4-1       up   infinite      0    n/a
```

(2) View the job queue

```
squeue
```

- JOBID: Job ID.
- ST: Status (R: Running; CF: Configuring; PD: Queueing).

```
[cloudam@master jobs]$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
                20     c-4-2  demo.sh  cloudam  R       0:29      1 c-4-2-worker0001
```

(3)View all job details

```
scontrol show jobs
```

```
[cloudam@master jobs]$ scontrol show jobs
JobId=20 JobName=demo.sh
   UserId=cloudam(1003) GroupId=cloudam(1003) MCS_label=N/A
   Priority=4294901756 Nice=0 Account=(null) QOS=normal
   JobState=RUNNING Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:16:15 TimeLimit=UNLIMITED TimeMin=N/A
   SubmitTime=2021-11-19T16:04:23 EligibleTime=2021-11-19T16:04:23
   AccrueTime=2021-11-19T16:04:23
   StartTime=2021-11-19T16:04:25 EndTime=Unknown Deadline=N/A
   SuspendTime=None SecsPreSuspend=0 LastSchedEval=2021-11-19T16:04:25
   Partition=c-4-2 AllocNode:Sid=iZ2zehs3cdd8rz787les3wZ:5927
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=c-4-2-worker0001
   BatchHost=c-4-2-worker0001
   NumNodes=1 NumCPUs=4 NumTasks=4 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   TRES=cpu=4,node=1,billing=4
   Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
   MinCPUsNode=1 MinMemoryNode=0 MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
   Command=/home/cloudam/jobs/demo.sh
   WorkDir=/home/cloudam/jobs
   StdErr=/home/cloudam/jobs/slurm-20.out
   StdIn=/dev/null
   StdOut=/home/cloudam/jobs/slurm-20.out
   Power=
   MailUser=(null) MailType=NONE
```

(4)Cancel the job with job number 20

```
scancel 20
```

2.How to submit assignments

(1)Submit jobs using sbatch batch mode

> The sbatch command can submit tasks to one or more computing nodes for parallel computing.

Some common options for the sbatch command:

parameter	function
-N	The number of nodes applied for
-p	Specify the computing node specifications, use sinfo to view the supported specifications
-n	Specify the number of tasks, that is, how many processes the parallel program runs
-c	The number of cpu cores used by each process

Reference running program: demo.sh

```
#!/bin/bash
sleep 6000
```

Submit an example:

- Start 8 parallel tasks using 2 4 core nodes.

  ```
  sbatch -N 2 -p c-4-1 -n 8 -c 1 demo.sh
  ```

- 1 4-core node starts 4 parallel tasks.

  ```
  sbatch -N 1 -p c-4-1 -n 4 -c 1 demo.sh
  ```

- 1 4-core node starts 4 parallel tasks.

  ```
  sbatch -p c-4-1 -n 4 demo.sh
  ```

- 1 4-core node starts 1 parallel task, which uses 4 cpu cores.

  ```
  sbatch -p c-4-1 -n 1 -c 4 dome.sh
  ```

- 2 4-core nodes start 2 parallel tasks, and each task uses 4 cpu cores.

```
sbatch-p c-4-1 -n 2 -c 4 dome.sh
```

(2)Submit jobs using salloc allocation mode

> The salloc command is used to allocate nodes. After the user obtains the allocated computing node, use ssh to enter the node to directly run the relevant computing program, which is mainly used to debug program execution.

Use steps (case):

Step1. Salloc applies for computing nodes;

```
salloc -N 1 -p c-8-2 &
```

Step2. SSH login to the assigned computing node;

```
ssh c-8-2-worker0001
```

Step3. Debug or run the program;

```
./demo.sh
```

Step4. Release the allocated node after the program is finished running;

```
scancel 3
```

Use of the Module System

> Module is an environment variable management tool. The BeiKun cloud HPC platform has installed many public software. Users only need to load the module to use the platform's software or dependent libraries. If they do not find the required software, they can contact customer service for installation.

1.Common commands

```
module avail or module av     #View the software available in the system
module spider or module sp    #Query all possible modules
module add or module load     #Load module
module rm or unload           #unload the module
module list or module li      #Display loaded modules
module purge                  #Uninstall all modules
module show                   #Display module configuration file
module swap or module switch  #replace module 1 with module 2
module help                   #Display help information
```

Note: It is recommended not to add multiple software modules at the same time, because there may be conflicts between different software. A better way is to module add one or a group of interdependent software. After the software runs, run module purge to clear the imported environment, and then import another or a group of interdependent software.

2.Usage Examples

View the existing software environment of the cluster

```
module avail
```

View gromacs software environment

```
module avail gromacs
```

Load GROMACS/2021-gromacs software environment

```
module add GROMACS/2021-gromacs
```

show all loaded environments

```
module list
```

Clear the loaded environment

```
module purge
```

Use of Conda Environment

> Conda is an open source package management system and environment management system for installing multiple versions of packages and their dependencies and easily switching between them. Conda supports Python, R, Ruby, Lua, Scala, Java, JavaScript, C/C++, FORTRAN and other languages.

1.Using the Conda environment in Cloudam

```
#Load Anaconda3
module add Anaconda3

#View Conda environment list
conda env list

#Load the specified virtual environment
source activate xxxx

#Exit the current environment
conda deactivate
```

2.Conda virtual environment

```
#Create environment
conda create -n xxxx

#Create a virtual environment named demo and specify the python version as 3.7
conda create -n demo python=3.7

#delete the environment
conda env remove -n xxxx
```

3.Conda package management

```

```

```
#Check if a package can be installed with conda
conda search numpy

#Installation package
conda install numpy

#Install the specified version package
conda install numpy=1.14

#View the packages installed in the current environment
conda list

#package update
conda update numpy

#delete package
conda remove -n demo numpy
```

4. Advanced operations

```
#List all images that save the target software
anaconda search <software>

#Specify the image of anaconda to install the version of cudatoolkit10.0
conda install -c anaconda cudatoolkit=10.0

#pip install the specified version of the package
pip install some-package==1.0.4

#pip install the latest version of the package
pip install some-package

#pip upgrade package
pip install --upgrade some-package

#pip uninstall package
pip uninstall some-package
```