



华南理工大学
South China University of Technology

科学计算公共服务平台

基础入门

- 一 平台概况
- 二 申请账户
- 三 登录集群
- 四 文件传输

- 五 提交作业
- 六 账户管理
- 七 平台演示
- 八 常见问题



华南理工大学

South China University of Technology

科学计算公共服务平台

一 平台概况



为什么需要科学计算公共服务平台？

- 需要大规模计算，超出笔记本电脑或工作站的处理能力；
- 可能有太多数据，需要海量存储和高速读写；
- 可能需要先进高效的 GPU 资源，抑或是大内存节点。

这些，都能在华南理工大学科学计算公共服务平台上实现。华南理工大学科学计算公共服务平台面向全校师生提供服务，支撑和催化学校的科研工作。

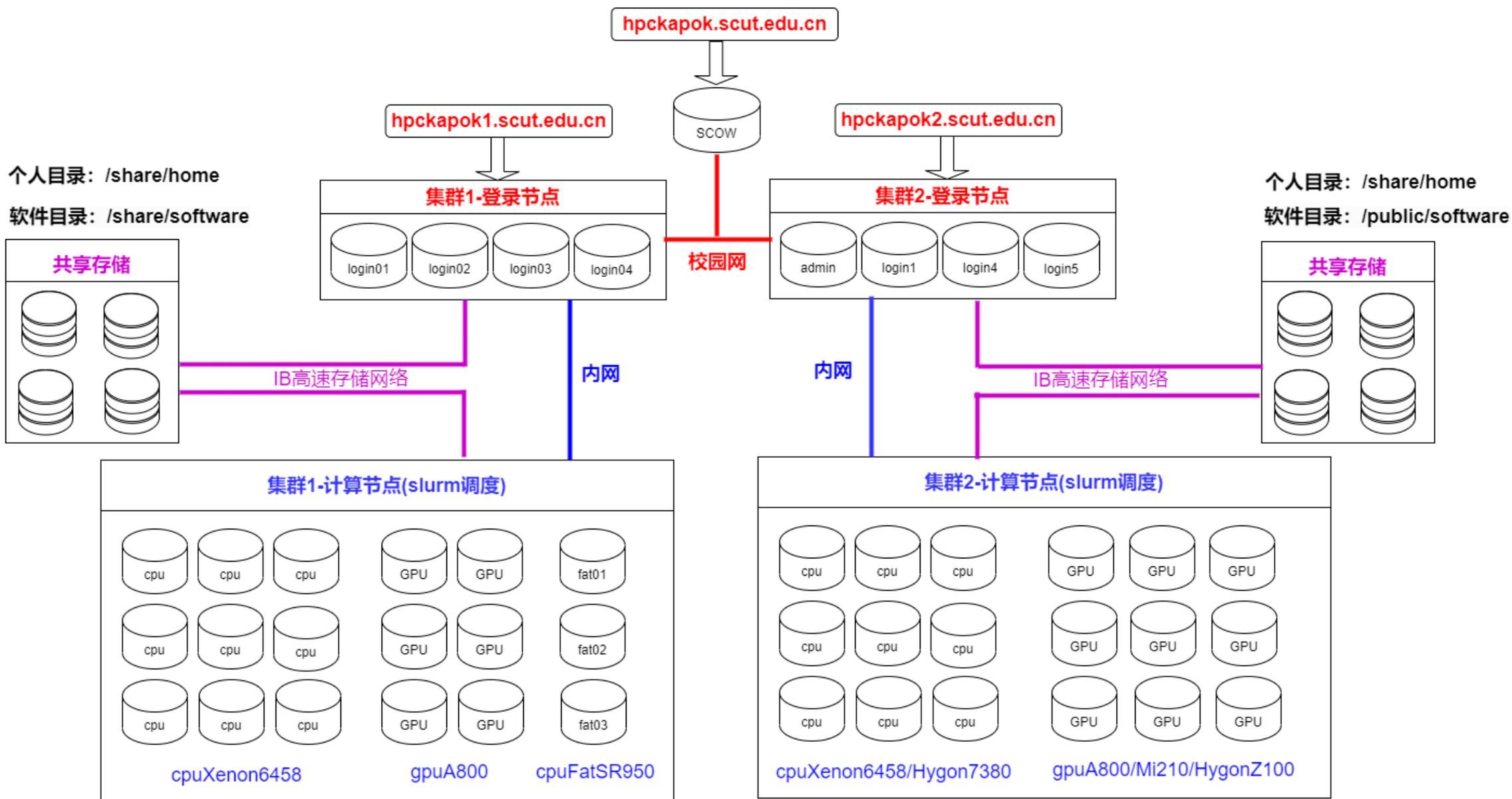
平台的主要功能：提供高性能科学与工程计算服务和AI计算服务。

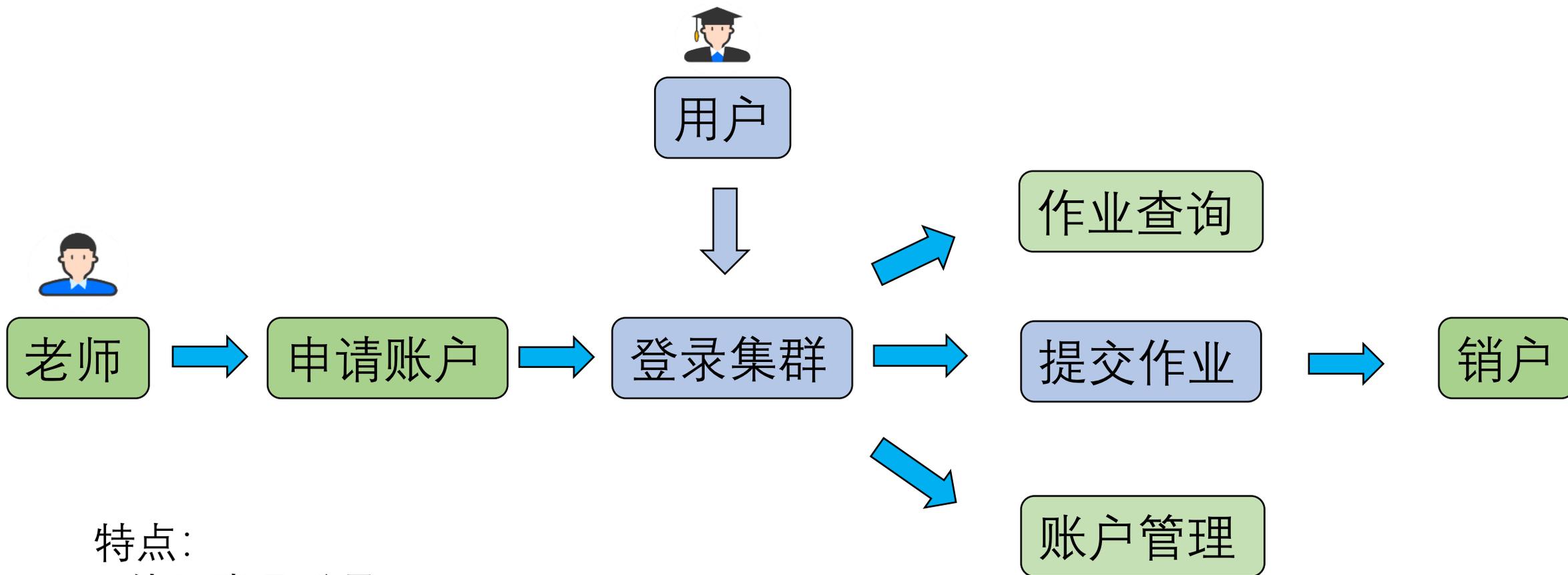
主要应用领域包括：数学、深度学习、新能源新材料等，涉及数学、统计学、力学、物理学、化学、生物学、生命科学、计算机科学等多学科科学研究及工程应用。

平台原理：利用集群中多台计算机协同运作，可以提供更高的计算性能和数据处理速度，适用于大规模数据处理、科学计算等领域。



节点类别	型号	规格
CPU节点	Lenovo SD650	2*32C Intel Xeon Gold
		16*32GB DDR5内存
GPU节点1	Lenovo SD650-N	2*18C Intel Xeon Gold
		4*A800 80GB GPU
		16*64GB DDR4内存
胖节点	Lenovo SR950	8*24C Intel Xeon Platinum
		96*64GB DDR4内存
登录节点	Lenovo SR630	2*32C Intel Xeon Gold
		16*32GB DDR5内存
CPU节点1	BX50 LP G50	2*32C Intel Sapphire Rapids 6458Q
		16*16GB DDR5内存
CPU节点2 (国产)	CX52 G40L	2*32C Hygon 7380 2.2GHz
		16*32GB DDR4内存
GPU节点1(nvidia)	X660 G45 LP	2*32C Intel 8358P 2.6G 11.2UPI 48M
		16*32GB DDR4内存
		NV HGX A800 8-GPU 80GB
GPU节点2(AMD)	X640 G40	2*32C Intel 8358P 2.6G 11.2UPI 48M
		16*32GB DDR4内存
		AMD Instinct MI 210*
GPU节点3 (国产)	X7840H0	2*32C Hygon 7380 2.2GHz
		16*32GB DDR4内存
		8* Hygon DCU Z100 32G
登录节点	R620 G40	2*32C Intel 8358P 2.6G 11.2UPI 48M
		16*32GB DDR4内存





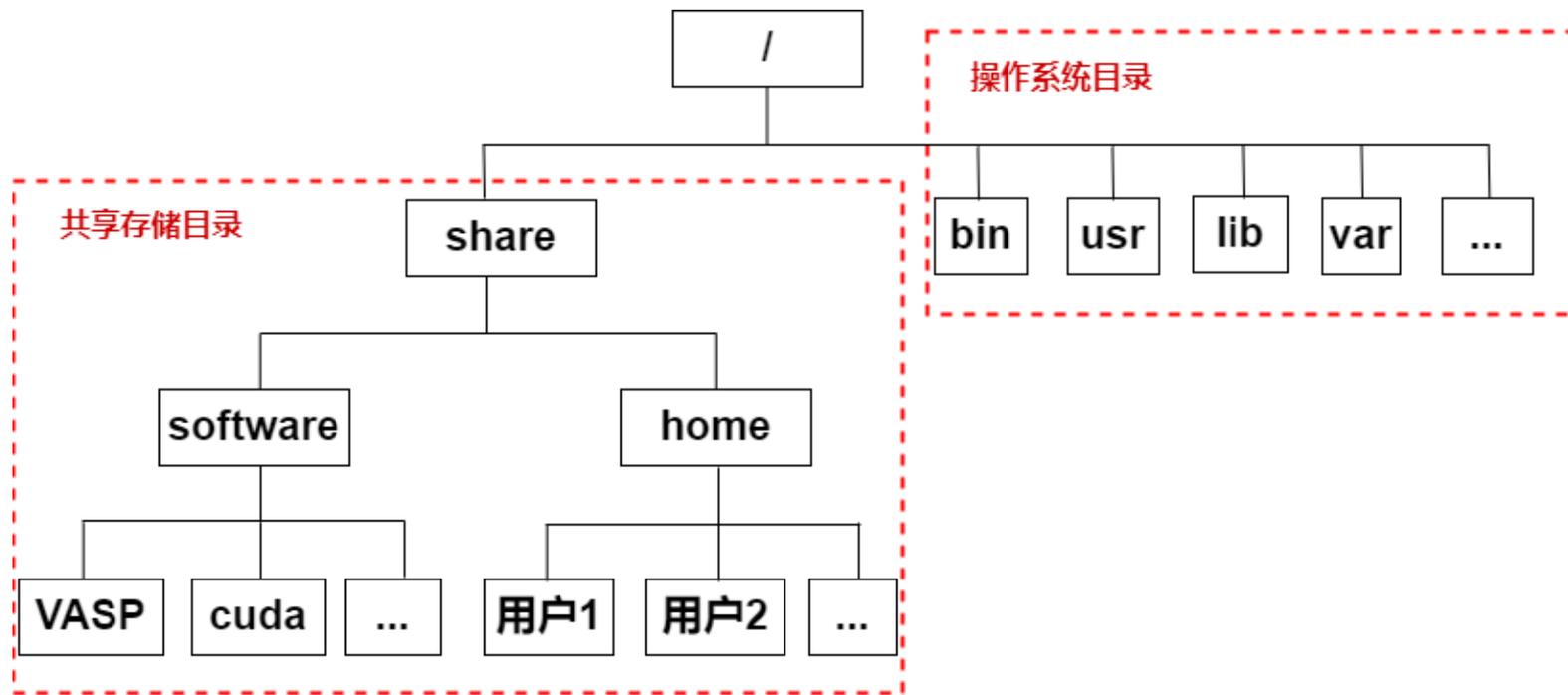
特点:

- 1.使用账号登录
- 2.服务器为Linux操作系统



操作系统(linux): centos8.4, rockylinux8.6

linux命令	功能
ls	列出目录内容
pwd	显示当前目录的绝对路径
cd	切换到指定目录
cp	拷贝文件或者目录
mkdir	创建目录
mv	移动或者重命名文件或目录
rm	删除文件或目录
more	查看文件内容
tail	动态查看最后几行内容



绝对路径: /share/software/VASP

相对路径: ./VASP



华南理工大学

South China University of Technology

科学计算公共服务平台

二 申请账户



- 1、打开华南理工大学办事大厅<https://ehall.scut.edu.cn/>，使用申请人的统一认证账号登录。搜索服务“科学计算公共服务平台账户开户”，点击进入开户流程。
- 2、仔细阅读并确认注意事项，补充新增成员信息、支撑项目信息，上传所有需开通账号的用户签名后的责任书，确认相关信息后提交开户申请。
- 3、平台服务团队在收到申请后，将为所有用户建立**账号**和一个**账户**，以邮件方式通知申请人的用户名、密码以及登录信息。

• 账号：

基本作用：登录平台使用的id，用户名为**统一认证账号或者学号**。

负责人账号：为负责老师，账户拥有者，登录平台，账户管理。

成员账号：为学生或老师，登录平台，进行计算。

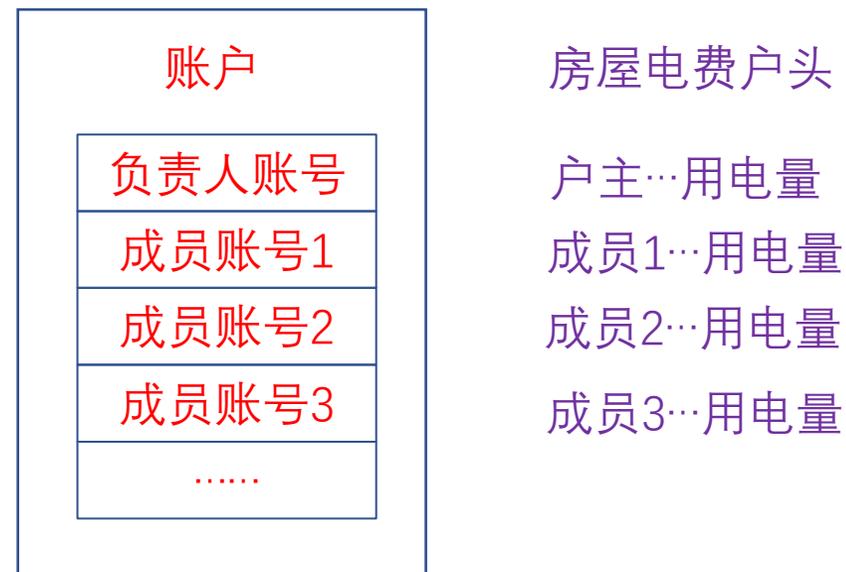
• 账户：

基本作用：

标识团队，记录作业数据，

账户拥有者为老师，可进行账户内成员管理、查看成员作业。

一个账号可属于多个账户。





华南理工大学

South China University of Technology

科学计算公共服务平台

三 登录集群



Web访问

- 主门户(scow): <https://hpckapok.scut.edu.cn>
- 集群1门户 (hpckapok1) : <https://hpckapok1.scut.edu.cn>
- 集群2门户 (hpckapok2) : <https://hpckapok2.scut.edu.cn>

ssh访问

- 集群1 (hpckapok1) : hpckapok1.scut.edu.cn (202.38.252.202, 202.38.252.203, 202.38.252.204, 202.38.252.205) (账号密码访问)
- 集群2 (hpckapok2) : 202.38.252.210, 202.38.252.211 (登录集群2门户后下载密钥登录访问)

Windows下的工具软件: xshell, MobaXterm等。Linux下工具软件: openssh

远程桌面(VNC)

- web登录<https://hpckapok.scut.edu.cn>后, 点击“门户”---“桌面”---“新建桌面”, 点击“启动”
- 使用mobaXterm等客户端工具, 输入ssh访问的ip地址, 选择xfce4桌面



登录**计算节点**;

需要ssh登录节点后，再ssh登录到计算节点，命令如下：

```
$ ssh nodename
```

注意：**只能登录有作业运行的节点**

实现功能

A .Web访问

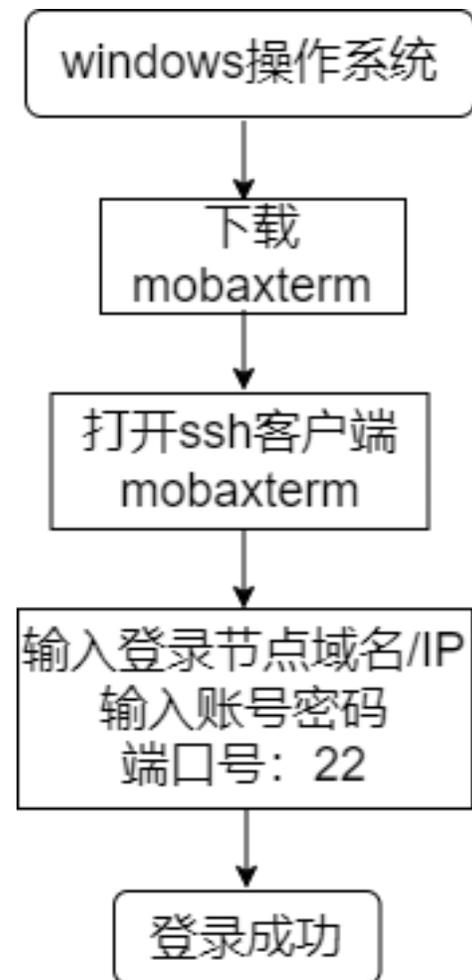
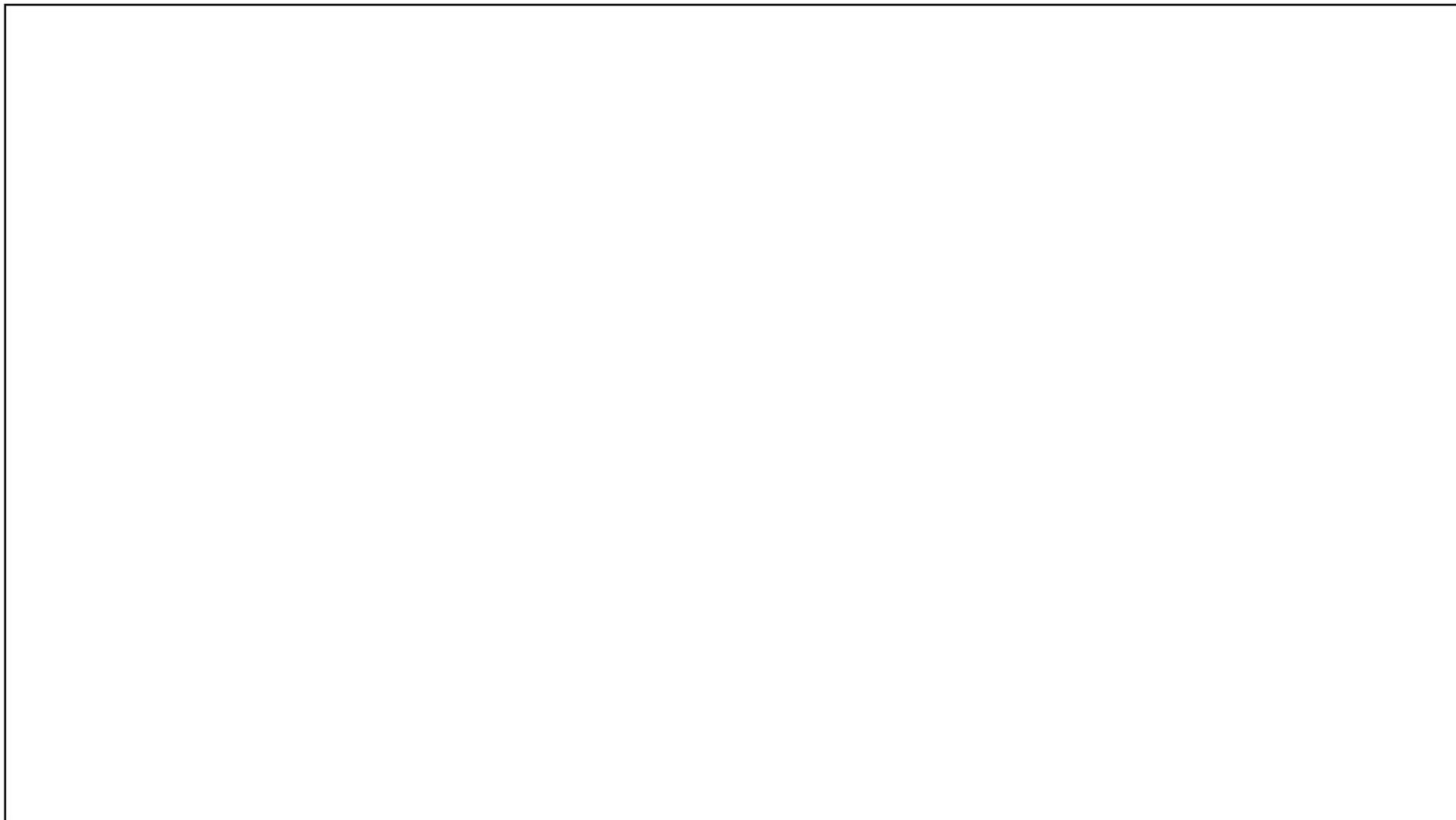
- 1.提交作业：命令行方式、web方式、可视化方式（交互式应用，远程桌面）
- 2.文件管理(上传下载)
- 3.作业查询
- 4.账号管理

B. ssh访问

1. 提交作业：命令行方式
2. 文件管理(上传下载)

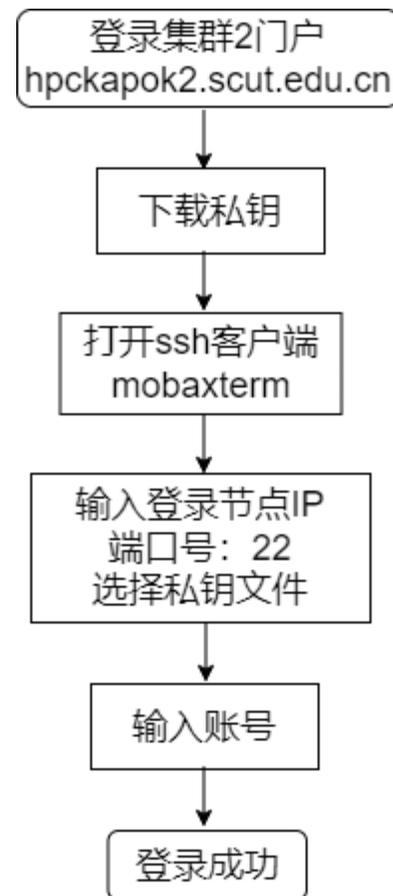
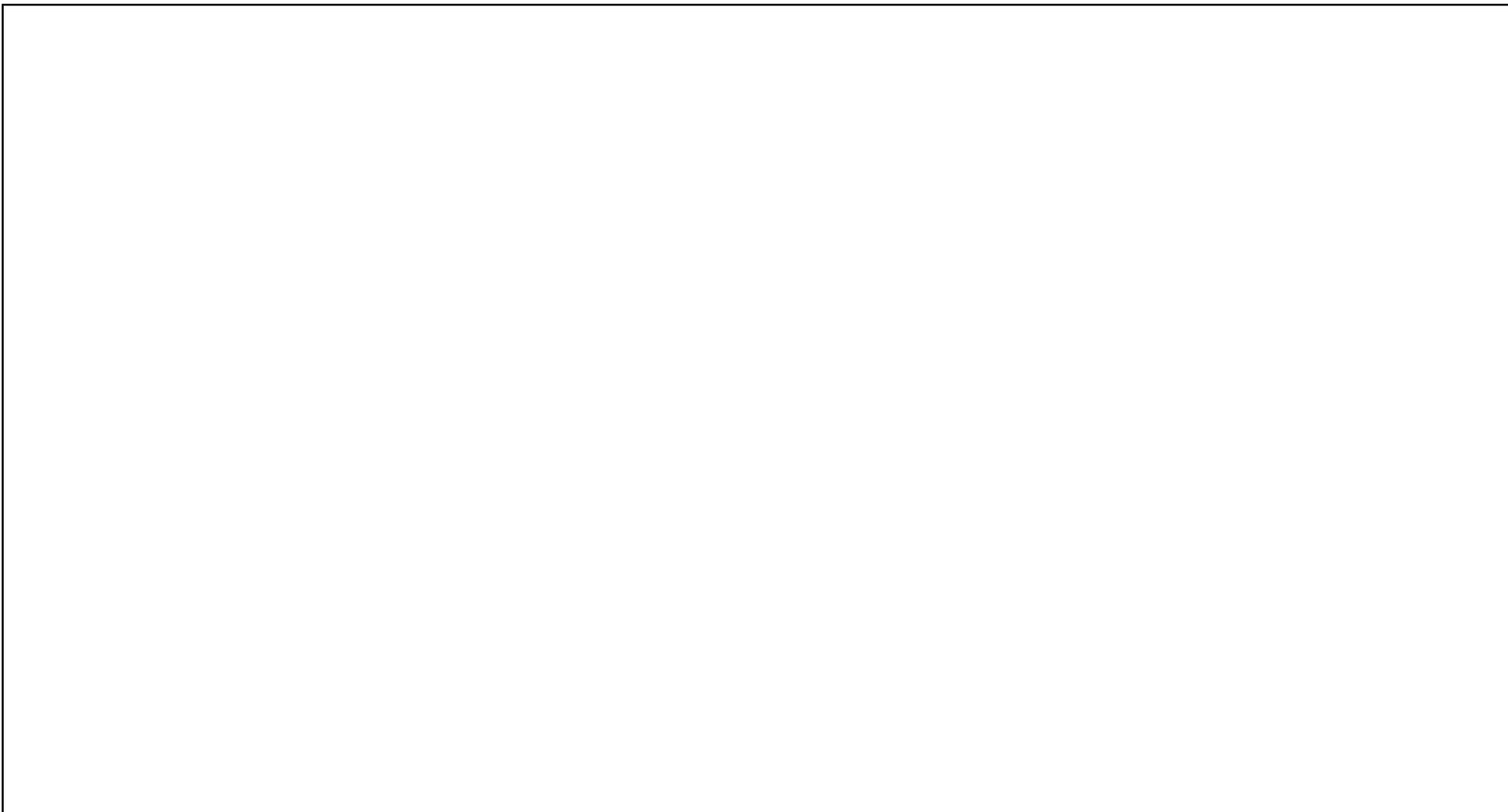


ssh登录集群1演示





ssh登录集群2演示





华南理工大学

South China University of Technology

科学计算公共服务平台

四 文件传输



个人目录: /share/home/username

个人程序和数据存放位置

文件传输

- 门户(文件管理)
- ssh客户端: windows(winscp, MobaXterm等), linux(scp等)
- 其他: wget, git等
- 数据量较大, 可与我们联系

➤ 集群1传输到集群2(在集群2上执行):

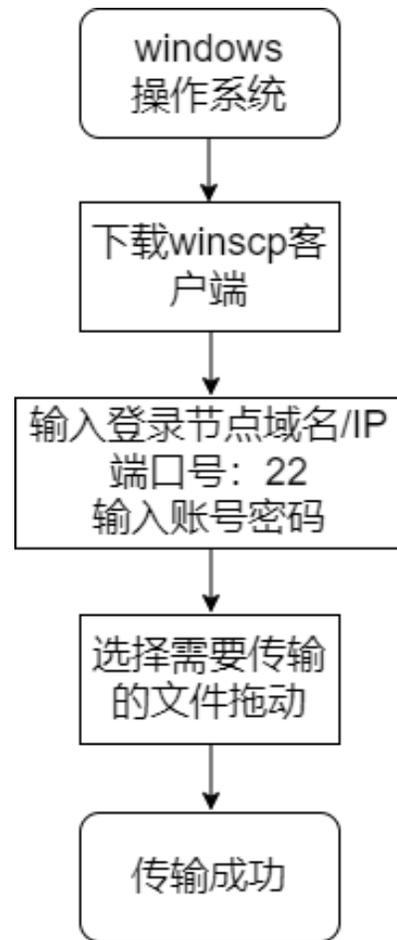
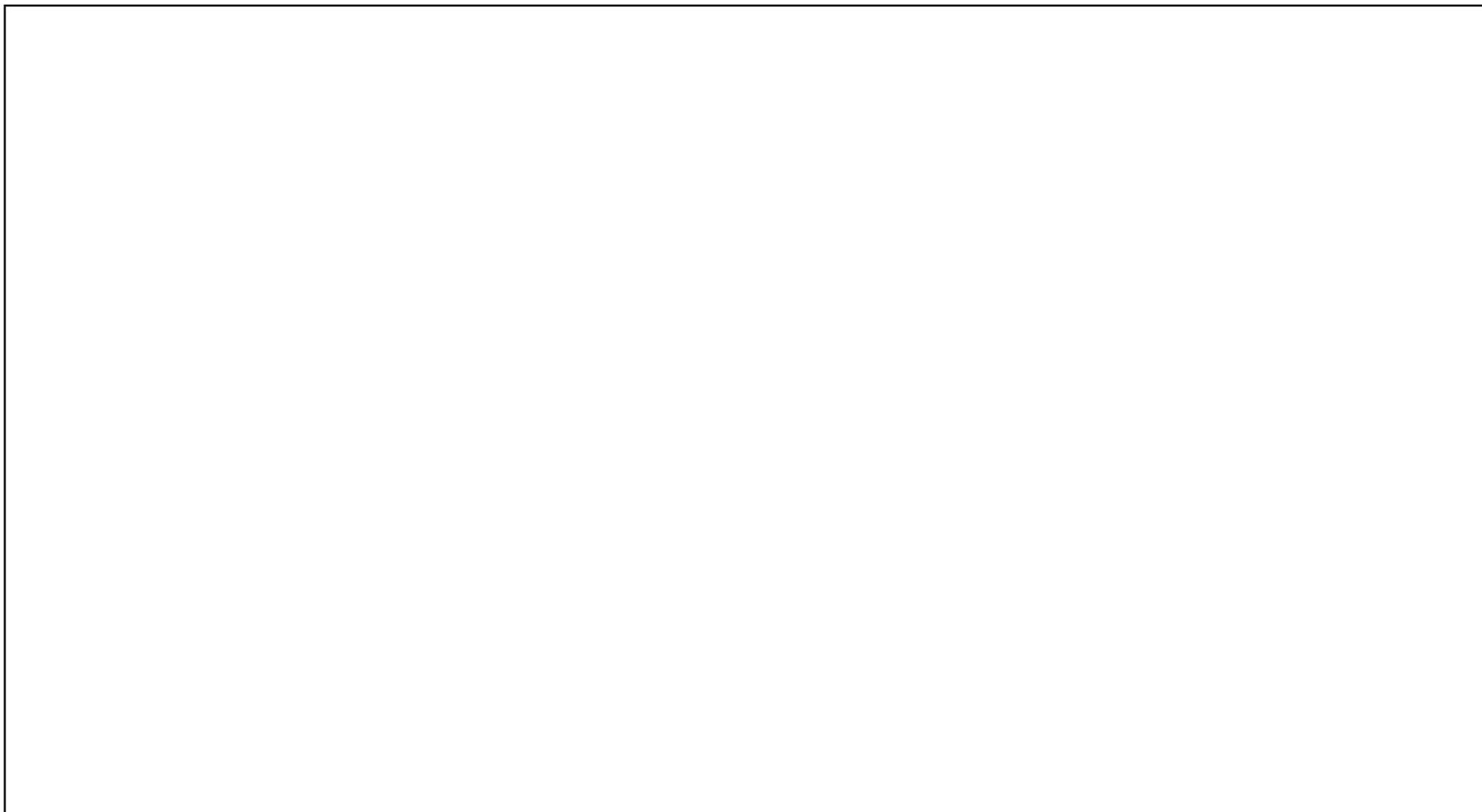
```
$ scp -r username@remote:/path/remotefile_dir /local/destination/
```

➤ 集群2传输到集群1 (在集群2上执行):

```
$ scp -r localfile_dir username@remote:/path/to/destination/
```



ssh客户端传输文件演示





华南理工大学

South China University of Technology

科学计算公共服务平台

五 提交作业



提交作业的方式:

- 命令行方式提交: ssh客户端, 门户网站的shell
- Web方式提交: 门户网站
- 可视化方式提交: 门户网站(交互式应用, 桌面)



命令行方式:

- 软件使用
- 软件安装
- Slurm调度系统
- 命令行方式提交作业



安装软件目录： /public/software,/share/software

编译器： gcc, intel(oneapi), nvhpc

并行库： openmpi, intelmpi(oneapi), mpich

应用软件：

人工智能(tensorflow, pytorch, paddlepaddle)

生物信息(AlphaFold2, Megahit)

建模和仿真(Ansys, Materials Studio, cst studio suite, comsol, wrf)

分子动力学计算化学(lammps , namd, gaussian, gromacs, vasp, cp2k)

…等

使用方法

- module模块： 加载平台已有软件
- 容器： singularity(集群1和集群2), docker(集群2)
- conda



Module模块命令

软件安装到自定义的目录后，并不能直接使用，需要将软件的可执行文件路径等添加到对应的环境变量后才能使用。module则是一款环境变量管理工具，通过module实现软件环境变量的管理，快速加载和切换软件环境。集群安装了常用的一些软件和库，可通过module进行加载使用。

module 命令	功能
module load [MODULE]	加载模块
module unload [MODULE]	卸载模块
module av	列出所有模块
module av keyword	列出名称中含有keyword的模块
module list	列出所有已加载的模块
module show [MODULE]	列出模块的信息，如路径，环境变量等



Module模块命令

\$ module av cuda



```
[zq@login04 ~]$ module av cuda
```

cuda/10.2	cuda/11.8 (L)	nvhpc/nvhpc-hpcx-cuda11/24.1
cuda/11.3.0	cuda/12.3 (D)	nvhpc/nvhpc-hpcx-cuda12/24.1

Where:
D: Default Module
L: Module is loaded

\$ nvcc --version
\$ module load cuda/10.2
\$ nvcc --version



```
[zq@login04 ~]$ nvcc --version
```

nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2022 NVIDIA Corporation
Built on Wed_Sep_21_10:33:58_PDT_2022
Cuda compilation tools, release 11.8, V11.8.89
Build cuda_11.8.r11.8/compiler.31833905_0

```
[zq@login04 ~]$ module load cuda/10.2
```

The following have been reloaded with a version change:
1) cuda/11.8 => cuda/10.2

```
[zq@login04 ~]$ nvcc --version
```

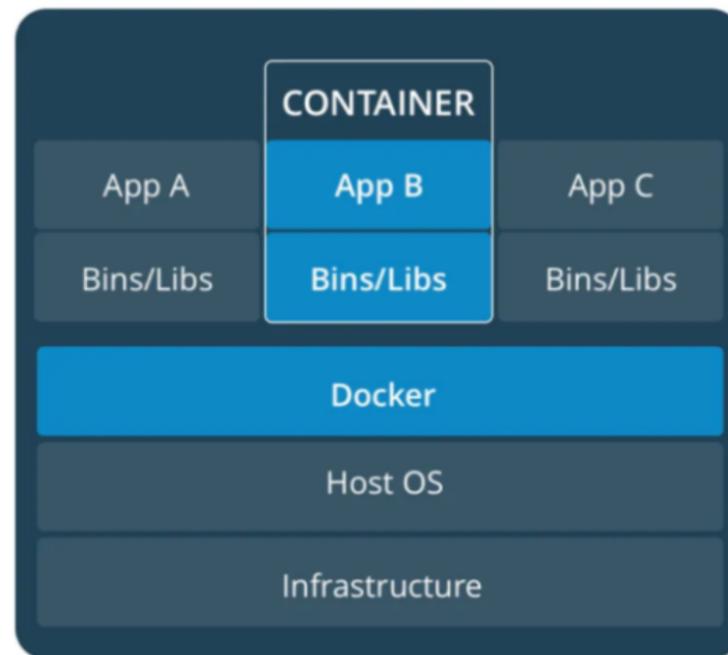
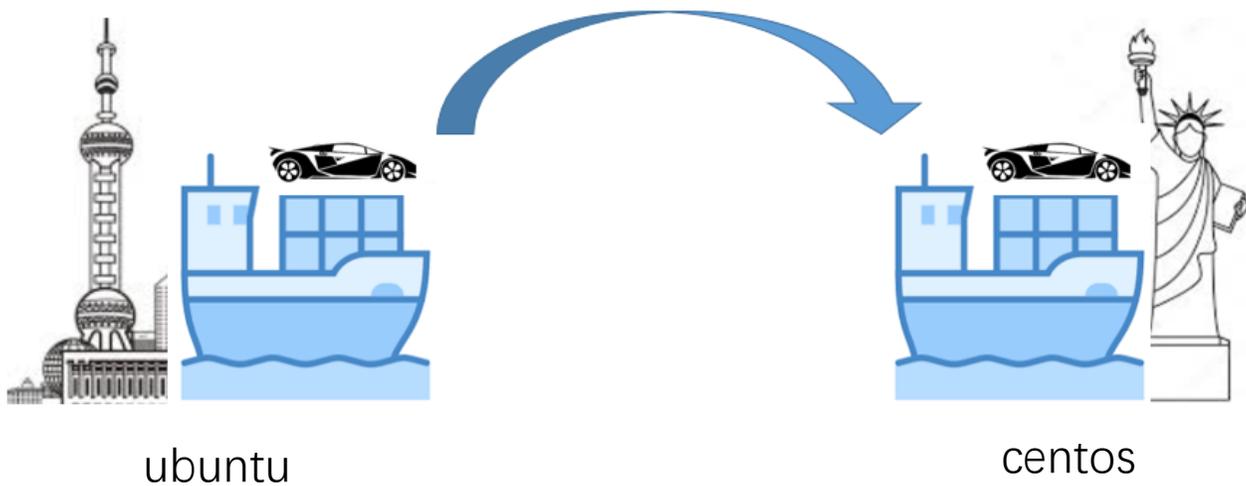
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_19:24:38_PDT_2019
Cuda compilation tools, release 10.2, V10.2.89



容器

容器是一种Linux上广为采用的应用封装技术，它将可执行程序与依赖库打包成一个**镜像**文件，启动时与宿主节点共享操作系统内核。由于镜像文件同时携带可执行文件和依赖库，避免了因系统环境不匹配造成的兼容性问题，因此它能够从一台主机迁移到另外一台主机，还能在一个宿主Linux操作系统上支持多种不同的Linux发行版。

常见的容器技术有docker, runc, Apptainer/Singularity, LXC, podman等。





docker容器

由于docker容器运行需要root权限，而平台普通用户无法获取root权限，所以docker不支持在命令行运行，只有在集群2的hpckapok2门户网上可以运行，后面web提交作业这一部分再讲述。

Singularity容器

singularity 的高性能计算容器技术，相比Docker等在云计算环境中使用的容器技术，Singularity 支持root用户和非root用户启动，容器启动前后，用户上下文保持不变，用户权限在容器内部和外部都是相同的。

```
$ singularity pull centos8.sif docker://centos:centos8 #拉取centos8镜像
$ singularity pull docker://cp2k/cp2k:latest #拉取cp2k镜像
$ singularity run my-container.sif #运行容器
$ singularity exec my-container.sif my-command #执行容器内命令
$ singularity shell my-container.sif #交互式运行
```

<https://sylabs.io/docs/>



conda

conda是一个包管理工具及python, R语言的环境管理工具, 可以用来管理python包, 相当于pip的升级版, 可以创建虚拟环境从而可以在一台终端创建多个python版本或python软件包及其依赖环境。conda的发行版本有miniconda、anaconda、Miniforge等, 关系类似于linux的发行版本包括ubuntu/centos/安卓等。

miniconda是conda的一个小型发行版本。它只包含conda, python, 少量依赖包, 以及少量工具如pip、zlib。**anaconda**是conda的一个大型发型(安装)版本。它包含conda, conda-build, python, 250+预安装的用于科学计算的包及其依赖, 包括SciPy, NumPy等等。

```
$ module load anaconda/3-2023.09          #加载已经安装的anaconda
$ conda list [-n env_name]                #查看当前环境(指定环境)下安装的包
$ conda env list                           #查看当前存在哪些虚拟环境
$ conda create -n env_name [python=<version>] #创建环境, 并指定python版本, 或者安装包等
$ source activate env_name                 #激活虚拟环境
$ conda deactivate                          #退出虚拟环境
$ conda search pkg_name                    #查找安装包
$ conda remove pkg_name                    #删除指定的安装包
$ conda remove -n env_name --all          #删除指定环境
```



举例(cp2k)

使用module方式:

```
$ module load cp2k/2024.1 #加载cp2k
```

```
$ cp2k.psmtp -v #查看cp2k版本和编译选项
```

使用singularity镜像:

```
$ singularity run /share/software/images/cp2k/cp2k_latest.sif cp2k -v
```

使用conda安装的版本:

```
$ module load anaconda/3-2023.09 #加载anaconda
```

```
$ source activate cp2k202401 #激活cp2k环境
```

```
$ cp2k.ssmtp -v
```

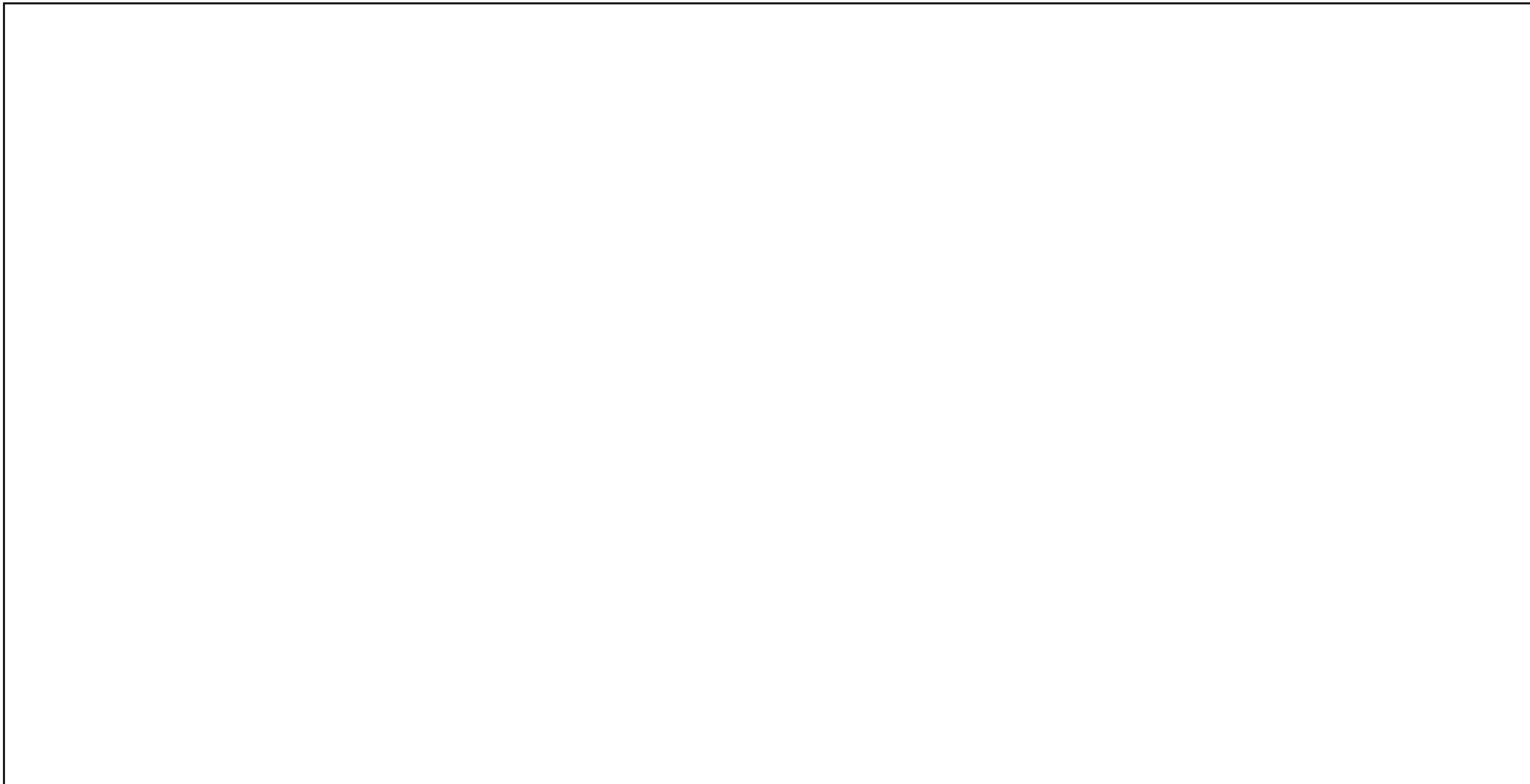


华南理工大学
South China University of Technology

科学计算公共服务平台

软件使用

软件使用演示





软件安装

- Conda: Python/R/生信/Perl应用
- 容器: 制作容器
- 源代码编译安装

.....

- 1.只能安装到个人目录。其他目录没有写的权限。
- 2.如果是通用的软件，我们可以在平台全局安装部署。
- 3.Linux系统上的软件通常会依赖第三方库如mpi, fftw等，这些第三方库基本都已经平台上部署安装完毕。可以用module av查看，使用module load或者export加载。
- 4.普通用户不能使用sudo，对于需要root权限才能安装或使用的软件程序，可以使用容器，如果确实需要，可以联系我们协助。



Conda方式安装

```
$ module load anaconda/3-2023.09 #加载anaconda
```

```
$ conda create -n torch-env -y #创建虚拟环境
```

```
$ source activate torch-env # 进入上一步创建好的环境
```

```
$ conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c  
pytorch -c nvidia
```

```
# 安装相应软件
```

```
$ conda deactivate
```

```
# 退出当前的虚拟环境
```

普通用户无法在平台已部署的anaconda默认目录下创建虚拟环境,会提示“NoWritablePkgsDirError: No writeable pkgs directories configured.

 -/share/software/anaconda3/pkqs”,

因为相应目录没有写的权限。可以:

1.个人目录下创建.condarc文件, 添加如下内容:

```
auto_activate_base: false
```

```
pkgs_dirs:
```

```
- /share/home/$USERNAME/.conda/pkgs
```

2.自行安装miniconda/anaconda



构建singularity镜像(本地)

```
# singularity build --sandbox ubuntu docker://ubuntu:latest #使用ubuntu系统为系统基本镜像
Singularity> apt-get -y update
Singularity> apt -y install python3
Singularity> apt -y install python3-pip
Singularity> pip3 install torch torchvision torchaudio --index-url
https://download.pytorch.org/whl/cpu
# singularity build ubuntu.sif ubuntu #打包为新的镜像
INFO: Starting build...
INFO: Creating SIF file...
INFO: Build complete: ubuntu.sif
# singularity exec ubuntu.sif python3 -c "import torch;x = torch.rand(5, 3);print(x)" #运行
tensor([[0.2580, 0.4027, 0.4330],
        [0.0902, 0.7048, 0.0461],
        [0.4692, 0.7896, 0.4254],
        [0.9736, 0.9152, 0.8503],
        [0.3017, 0.8254, 0.0925]])
```

构建镜像方式:

- 1.拉取(pull)已有的镜像
- 2.Sandbox方式自行安装
- 3.使用def文件构建
- 4.先创建docker镜像再转换



源代码方式安装

```
$ wget http://fftw.org/fftw-3.3.10.tar.gz
```

```
$ tar -xf fftw-3.3.10.tar.gz
```

```
$ cd fftw-3.3.10
```

```
$ CC=gcc CXX=g++ FC=gfortran ./configure --prefix=${HOME}/fftw/3.3.10
```

```
$ make
```

```
$ make install
```

以官方安装文档为准



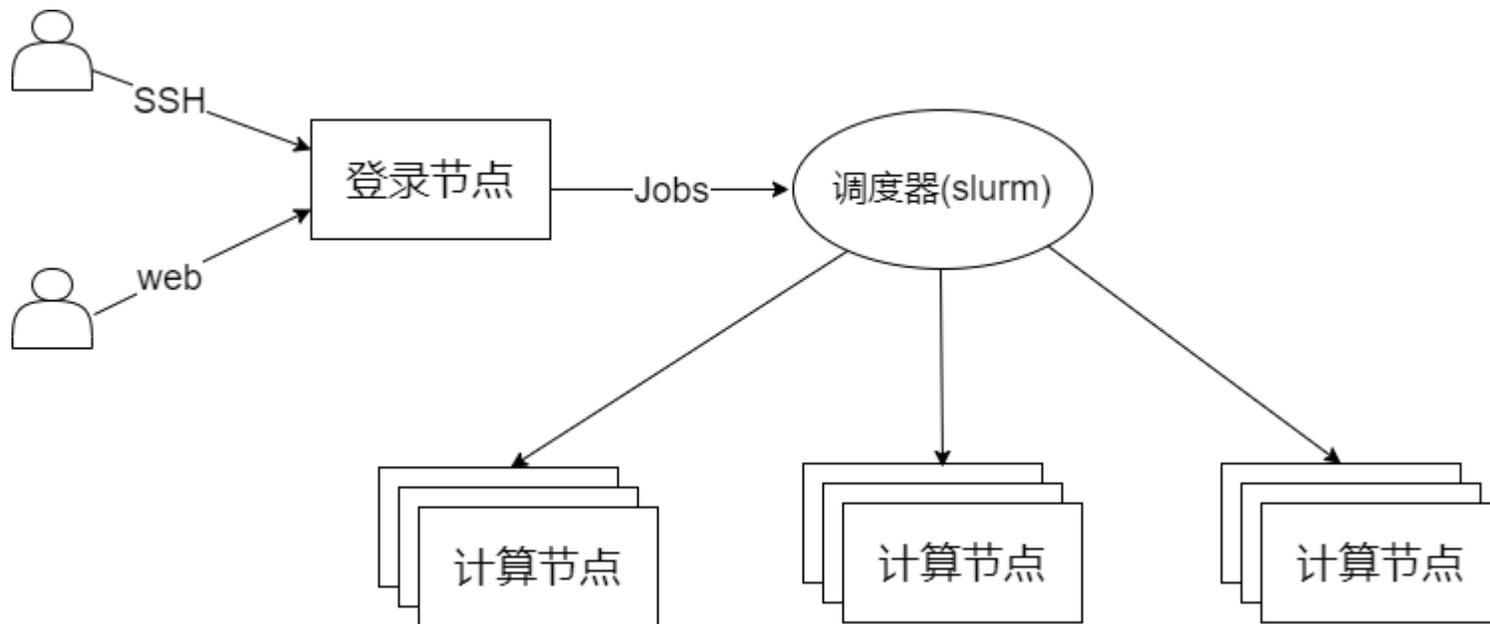
Slurm资源管理与作业调度系统

主要功能

- 资源管理:节点/cpu/内存/gpu等
- 提交作业:分区/并行等
- 作业监控:作业状态/资源使用情况
- 作业记录:历史记录

slurm的术语

- 节点 (Node) : 计算节点/登录节点
- 作业 (Job) : 排队/运行/失败/完成
- 队列 (Partition) : 节点列表/作业大小/作业时长/用户列表





slurm命令	功能	举例
sinfo	查看集群状态	<p>sinfo 查看所有信息</p> <p>sinfo -N --state=idle 查看状态(state)为可用(idle)的节点信息</p> <p>sinfo --partition=cpu 查看队列名为cpu的信息</p> <p>节点状态: drain(节点故障), alloc(节点在用), idle(节点可用), down(节点下线)</p>
squeue	排队作业状态	<p>squeue -l 查看作业排队细节</p> <p>squeue --state=R 查看状态为运行中的作业</p> <p>作业状态: R(Running), PD(Pending), completing(CG), completed(CD), Failed, Cancelled, Node_fail</p>
sbatch	提交作业	<p>sbatch jobscript.slurm 提交文件名为jobscript.slum的作业脚本</p>
scontrol	查看或设定slurm作业、分区、节点等状态	<p>scontrol show JOB_ID 查看作业ID为JOB_ID的作业信息</p>
sacct	查看作业记录	<p>sacct --state=CD 查看状态为完成(state为CD)的作业</p> <p>sacct -S YYYY-MM-DD 查看在指定时间(YYYY-MM-DD)后的所有作业</p>
scancel	取消作业	<p>scancel JOB_ID 取消作业ID为JOB_ID的作业</p>



- scontrol show partition查看分区信息:

主门户	集群	队列/分区	单节点配置			节点数量
			CPU	GPU	内存	
scow(hpckapok)	集群1(hpckapok1)	cpuXeon6458	64核	0	512G	320
		gpuA800	36核	4卡	1024G	32
		gpuV100	32核	4卡	192G	9
		cpuFatSR950	192核	0	6144G	3
	集群2(hpckapok2)	cpuXeon6458	64核	0	256G	195
		cpuHygon7380	64核	0	512G	16
		gpuA800	64核	8卡	512G	20
		gpuMi210	64核	8卡	512G	3
		gpuHygonZ100	64核	8卡	512G	5



- sinfo:

```
[zq@login04 ~]$ sinfo
PARTITION      AVAIL  TIMELIMIT  NODES  STATE  NODELIST
cpuXeon6458*   up     infinite   1      comp  c02n02
cpuXeon6458*   up     infinite   46     drain* c11n[01-12],c12n04,c13n[05-12],
cpuXeon6458*   up     infinite   128    down*  c17n[05-10,12],c18n[01-04],c19n
,c26n[01-12],c27n[01-12],c28n[01-12],c29n[05-12],c30n[01-02]
cpuXeon6458*   up     infinite   2      mix   c01n01,c25n07
cpuXeon6458*   up     infinite   140    alloc  c01n[02-12],c02n[01,03-12],c03n
0n[01-12],c12n[01-03],c13n[01-04],c16n[01-12],c17n[01-04,11],c25n[06,08-09]
cpuXeon6458*   up     infinite   1      idle  c06n03
gpuA800        up     infinite   23     down*  g01n[05-06],g02n[05-06],g03n[01
gpuA800        up     infinite   6      mix   g01n[01-03],g02n[02-04]
gpuA800        up     infinite   1      idle  g01n04
```

- squeue:

```
JOBID PARTITION  NAME      USER ST      TIME  NODES  NODELIST(REASON)
45170 cpuXeon64  job-2024 l      R 9-15:04:09      1 c03n01
45232 cpuXeon64  job-2024 l      R 9-03:45:48      1 c06n03
48532 cpuXeon64  f8-30000 t      R      1:34:11      1 c17n11
48554 cpuXeon64  JYQ_CK 2021  c      R      29:45      1 c25n06
```



- scontrol show node nodename 显示节点硬件信息

```
[zqi@login04 ~]$ scontrol show node g01n01
NodeName=g01n01 Arch=x86_64 CoresPerSocket=1
  CPUAlloc=14 CPUEfctv=36 CPUTot=36 CPULoad=5.45
  AvailableFeatures=(null)
  ActiveFeatures=(null)
  Gres=gpu:4
NodeAddr=g01n01 NodeHostName=g01n01 Version=22.05.10
OS=Linux 4.18.0-372.32.1.el8_6.x86_64 #1 SMP Thu Oct 27 15:18:36 UTC 2022
RealMemory=1000000 AllocMem=0 FreeMem=1001813 Sockets=36 Boards=1
State=MIXED ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
Partitions=gpuA800
BootTime=2024-04-23T11:18:13 SlurmdStartTime=2024-04-23T11:18:48
LastBusyTime=2024-04-23T16:14:13
CfgTRES=cpu=36,mem=1000000M,billing=36,gres/gpu=4
AllocTRES=cpu=14,gres/gpu=2
CapWatts=n/a
CurrentWatts=0 AveWatts=0
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```



slurm提交作业的三种方式

• sbatch 后台提交作业

sbatch 命令采用批处理方式运行作业，sbatch 命令在脚本正确传递给作业调度系统后立即退出，同时获取到一个作业号。sbatch 命令没有屏幕输出，默认输出日志为提交目录下的 slurm-JOBID.out 文件，可以使用 tail -f slurm-JOBID.out 实时查看日志，其中 JOBID 为作业号。

编写脚本 job.slurm，内容如下：

```
#!/bin/bash
#SBATCH -N 2          #2个节点
#SBATCH --ntasks-per-node=8  #单节点8个进程(并行)
#SBATCH -c=4          #每个进程4个cpu(多线程)
#SBATCH --gres=gpu:1  #申请一个gpu卡
#SBATCH -t 60         #限时60分钟
python myscript.py    #运行程序
```

或者编写如下的脚本：

```
#!/bin/bash
srun -N 2 -n 64 -c 4 -t 60 python myscript.py
```

然后在命令行执行 `sbatch -p gpuA800 job.slurm` 就可以提交作业。

参数	含义
-n, --ntasks=<number>	任务总数
-p [partition]	作业队列
--job-name=[name]	作业名
--output=[file_name]	标准输出文件
--error=[file_name]	标准错误文件
--time=[dd-hh:mm:ss]	作业最大运行时长
-w, --nodelist=[nodes]	偏好的作业节点
-c, --cpus-per-task	单任务cpu核数
--gres=<list>	指定资源列表



slurm提交作业的三种方式

- **salloc分配模式作业提交**

salloc 命令用于申请节点资源, 过程如下:

```
$ salloc -N 1 -p gpuA800 #申请1台服务器资源
$ squeue                 #执行squeue查看分配到的节点资源, 比如分配到 gpu1
$ ssh gpu1              #登陆到所分配的节点
$ python myscript.py   #登陆节点后可以执行需要的提交命令或程序
$ scancel JOBID        #作业结束后, 执行 释放分配模式作业的节点资源
```

- **srun 交互式提交作业**

srun [options] command 命令属于交互式提交作业, 有屏幕输出, 但容易受网络波动影响, 断网或关闭窗口会导致作业中断。一般仅在调试程序时使用此方式提交作业。

命令示例:

```
$ srun -p gpuA800 -w gpu[1-2] -N 2 -n 80 -t 20 python myscript.py
```

```
-p gpuA800 指定提交作业到 gpuA800 队列
-w gpu[1-2] 指定使用节点 gpu1, gpu2
-N 2 指定使用 2 个节点
-n 80 指定进程数为 80
-t 20 指定作业运行时间限制为 20 分钟
```



ssh直接访问登录节点：

- 集群1（hpckapok1）：hpckapok1.scut.edu.cn（202.38.252.202，202.38.252.203，202.38.252.204，202.38.252.205）（账号密码访问）
- 集群2（hpckapok2）：202.38.252.210，202.38.252.211（登录集群2门户后下载密钥登录访问）

工具：

1. 网站门户上使用shell
2. windows下可以用Xshell，MobaXterm，putty等客户端软件
3. 远程桌面(VNC)打开终端



The screenshot shows the SCOW web interface. At the top, there is a navigation bar with icons for '仪表盘' (Dashboard), '作业' (Jobs), 'Shell', '桌面' (Desktops), '交互式应用' (Interactive Applications), and '文件管理' (File Management). The 'Shell' menu is highlighted with a red box and labeled '1'. A dropdown menu is open, showing a list of login nodes: 'login02', 'login04', 'login01', and 'login03'. The 'login04' node is highlighted with a red box and labeled '2'. Below the navigation bar, the main content area is titled '登录节点上的桌面' (Desktops on login nodes). It features a form with '集群' (Cluster) set to 'hpckapok1' and '登录节点' (Login Node) set to a dropdown menu. There are buttons for '刷新' (Refresh) and '+ 新建桌面' (New Desktop). Below the form is a table with columns for '桌面ID' (Desktop ID), '桌面名称' (Desktop Name), '桌面类型' (Desktop Type), and '地址' (Address). The table is currently empty, with a message '暂无数据' (No data) at the bottom.

门户上的shell:



← → ↻ https://hpckapok.scut.edu.cn/she

以ID: zq... 连接到集群 hpckapok1 的 login03 节点

```
No Slurm jobs found on node.
Last login: Fri Mar 22 11:12:56 2024 from 202.38.2...
[zq...@login03 ~]$ cd case/lammps/lammps/lammpsJob1/
[zq...@login03 lammpsJob1]$ ls
conda.lammps.sh lammps.sh log.lammps M-11.data M-1.in slurm-40610.out
[zq...@login03 lammpsJob1]$ more lammps.sh
#!/bin/bash
#SBATCH --job-name lammps_job
#SBATCH --partition cpuXeon6458
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=20

module load oneapi/2024.0
module load lammps/2Aug2023

mpirun lmp intel cpu intelmpi -i M-1.in
[zq...@login03 lammpsJob1]$ sbatch -A a_nicoper lammps.sh
Submitted batch job 40611
[zq...@login03 lammpsJob1]$ squeue

```

JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST (REASON)
40611	cpuXeon64	lammps_j	zqliu	R	0:02	1	c03n02
40528	cpuXeon64	lammps	zqliu	P	19:12:52	1	g01n02
40595	cpuXeon64	lammps	zqliu	P	19:12:52	1	g01n02

```
[zq...@login03 lammpsJob1]$ scancel 40611
[zq...@login03 lammpsJob1]$
```

门户上的
shell:



```
$ cd case/lammps/lammps/lammpsJob1/
```

```
#切换到相应目录
```

```
$ ls #查看当前目录文件
```

```
conda.lammps.sh lammps.sh log.lammps M-11.data M-1.in slurm-40610.out
```

```
$ more lammps.sh #查看lammps.sh文件的内容
```

```
#!/bin/bash
```

```
#SBATCH --job-name lammps_job
```

```
#SBATCH --partition cpuXeon6458
```

```
#SBATCH --nodes=1
```

```
#SBATCH --ntasks-per-node=20
```

```
#以上申明申请一个计算节点，20个cpu核,分区名cpuXeon6458，作业名lammps_job
```

```
module load oneapi/2024.0
```

```
module load lammps/2Aug2023
```

```
mpirun Imp_intel_cpu_intelmpi -i M-1.in
```



```
$ sbatch lammmps.sh
```

```
Submitted batch job 40611
```

```
#此命令提交作业
```

```
$ squeue
```

JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST(REASON)
40611	cpuXeon64	lammmps_j	zqliu	R	0:02	1	c03n02

```
#此命令查看作业
```

```
$ scancel 40611
```

```
#此命令取消作业
```



- 主门户(scow): <https://hpckapok.scut.edu.cn>
- 集群1门户 (hpckapok1) : <https://hpckapok1.scut.edu.cn>
- 集群2门户 (hpckapok2) : <https://hpckapok2.scut.edu.cn>

如下方式提交:

- Web页面写入脚本
- Web页面启动容器



Web方式

主门户

1

2

3

4

```
module load oneapi/2024.0
#加载intel编译器
module load lammeps/2Aug2023
#加载lammeps软件
mpirun Imp_intel_cpu_intelmpi -i temp.2.in
#运行程序
```



华南理工大学

South China University of Technology

科学计算公共服务平台

Web方式

← → ↻ ⚠ 不安全 <https://hpckapok1.scut.edu.cn/#>

- 🏠 首页
- 📁 作业模板**
- 📄 作业列表
- 📊 报告
- ☁️ 云工具
- 📁 Lenovo Accelerated DL
- 🔄 workflow
- ⚙️ 管理

首页 > 作业模板 > Common Job



Common Job

Create a Lenovo LiCO common job

模板信息

* 作业名称

* 工作目录

模板参数

* 运行脚本

```
module load gnepi/2024.0
module load lammps/2Aug2023
mpirun imp_intel_cpu_intelmpi -i temp.2.in
```

资源选项

* 队列

📊 UP 📄 320 nodes 📦 20095 cores 🗨 UNLIM

集群1门户



Gridview x +

hpckapok2.scut.edu.cn/w

GRIDVIEW 工作空间 我的应用 我的作业 我的账单 我的数据

< 返回 BASE 模板提交

gpuMi210
空闲节点/核心: 2/128

cpuHygon7380
空闲节点/核心: 12/776

cpuXeon6458
空闲节点/核心: 3/192

gpuHygonZ100
空闲节点/核心: 1/123

gpuA800

cpuXeon6458 高级 载入模板 | 重置

作业名称: STDIN_0322_101627

核心/节点: 1

节点选择: 节点总数

节点: 1

运行时限: 72 小时

工作目录: [default]

脚本文件内容

```
1 module load oneapi/2024.0
2 module load lanaps/2Aug2023
3 apirun lap_intel_cpu_intelapi -i teap.2.in
4
```

新建 上传

public

- 文件名
- 模板

保存 提交作业

集群2门户



未结束的作业

所有作业

提交作业

作业模板

本用户未结束的作业

结束作业

集群: hpckapok1 作业ID:

作业ID	作业名	账户	分区	QOS	节点数	核心数	GPU卡数	状态	运行/排队时间	更多
40653	job-20240324-114249	a_nicoper	cpuXeon6458	normal	1	1	0	RUNNING	00:03	进入目录 详情 结束



华南理工大学

South China University of Technology

科学计算公共服务平台

Web方式

集群1门户
容器

The screenshot displays the LiCO web interface. On the left is a dark sidebar with navigation options: 首页, 作业模板 (highlighted), 作业列表, 报告, 云工具, Lenovo Accelerated DL, workflow, 管理, 容器镜像, 账单下载, VNC, API Key, and 运行环境. The main content area shows a breadcrumb '首页 > 作业模板' and a large blue header with the text '作业模板'. Below this is a filter section with '全部 (35)' and a '排序' dropdown. A grid of job template cards is shown, including 'General Job 通用', 'MPI MPI HPC', 'COMSOL HPC', 'Singularity 通用' (highlighted with a red box), 'Singularity MPI HPC' (highlighted with a red box), and 'Charliecloud MPI HPC'.



集群1门户
容器

• 作业名称
Singularity_04221641

• 工作目录
MyFolder/singularity 浏览

模板参数 ▾

• 容器镜像
caffe-cpu 系统

• 容器内运行脚本
sleep 100

资源选项 ▾

• 队列
cpuXeon6458

• 每节点的CPU核数
1

UP 318 nodes 18462 cores UNLIMITED UNLIMITED



集群1门户
容器

资源选项 ▾

* 队列
cpuXeon6458

* 每节点的CPU核数
1

每节点的GPU数

内存使用(MB)

最大运行时间 ⌚
24h eg.3d 4h 12m

通知选项 >

提交 预览

Preview Job

```
#SBATCH --partition=cpuXeon6458
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --time=0-24:00

echo job start time is `date`
echo `hostname`
file_path=tmp_Singularity_04221641_`uuidgen`.sh
cat > ${file_path}<< EOF
sleep 100
EOF
module try-load singularity
singularity exec -B /share/home/zqliu \
    \
    /share/lico/container/caffe-cpu.image bash ${file_path}
rm -f ${file_path}
echo job end time is `date`
```

取消 提交



集群1门户
容器

- 首页
- 作业模板
- 作业列表
- 报告
- 云工具
- Lenovo Accelerated DL
- workflows
- 管理
- 容器镜像**
- 账单下载
- VNC
- API Key
- 运行环境
- 发布

首页 > 管理 > 容器镜像

导入 构建

ID	名称	框架
30	paddlepaddle	paddlepaddle
29	cvat	other
28	tensorrt8	tensorrt
27	scikit	scikit
26	rstudio	rstudio
25	intel-pytorch-cpu	pytorch
24	pytorch	pytorch
23	jupyter-intel-py37	jupyter
22	jupyter-py38	jupyter
21	jupyter-py37	jupyter
20	jupyter-default	jupyter



集群2门户
容器

← → ↻ <https://hpckapok2.scut.edu.cn/> ☆

创世纪算未来

HPC服务

集群综合管理系统。对数据中心IT设备进行监控、管理，检测性能瓶颈和热点，资产、作业、云桌面一切尽在掌握之中；提供统一的调度管理，支持异构的调度核心，调度策略灵活，支持定制化和自定义开发。

AI服务

一站式人工智能解决方案。深度学习全流程覆盖，集成主流AI框架，容器化环境部署，支持多种操作界面。提供模型训练、自动调优、在线推理、内容共享、环境定制等多种功能。



集群2门户
容器

容器实例 我的镜像 镜像仓库 任务列表

创建容器

删除

停止

按名称搜索...



<input type="checkbox"/>	任务名称	框架版本	任务模式	规格	提交时间
--------------------------	------	------	------	----	------

暂无数据



容器实例 我的镜像 镜像仓库 任务列表

创建容器 删除 停止

按名称搜索...

全部类型

全部状态

任务名称	框架版本	任务模式	规格	提交时间	状态	操作
<input type="checkbox"/> Instances_240422 1138	vscode-tensorflow:tensorfl ow2.10-py3.8-cuda11.8	单实例	2 核心; 30.0G 内存; 0 加速器	2024-04-22 17:00:06	运行	    

进入命令行

停止作业

集群2门户:容器



集群2门户
容器

GRIDVIEW Notebook 数据管理 算法管理 模型训练 共享中心

容器实例 我的镜像 镜像仓库 任务列表

构建镜像 去镜像仓库克隆

CPU ubuntu18:18.04

🕒 2024-04-17 11:14:37

镜像类型 Base

GPU pytorch:pytorch1.13-py3.8-cuda11.8

🕒 2024-01-24 20:46:33

镜像类型 Base PyTorch

< 返回 | 构建镜像

* 名称

* 标签

描述 0/100

适用加速器

不使用加速器	DCU	GPU
---------------	-----	-----

选择镜像类型

Base	Jupyter	VS Code
RStudio	PyTorch	PyTorch Horovod
PyTorch Inference	TensorFlow	TF Horovod
TF Tuning	TF Inference	

展示图片

构建方式

镜像包	Docker Hub	光源	Dockerfile
-----	-------------------	----	------------

* 源镜像拉取命令

[去 Docker Hub 搜索镜像并获取拉取命令](#)

重置 构建



华南理工大学
South China University of Technology

科学计算公共服务平台

可视化方式

可视化方式提交:

- 交互式应用
- 远程桌面(VNC)



交互式应用





交互式应用

The screenshot shows the SCOW (SUPER COMPUTING ON WEB) web interface. The browser address bar shows the URL `https://hpckapok.scut.edu.cn/a`. The navigation menu includes '仪表盘', '作业', 'Shell', '桌面', '交互式应用' (highlighted), and '文件管理'. The left sidebar shows two project folders: 'hpckapok1' and 'hpckapok2', each with '已创建的应用' and '创建应用' options. The main content area is titled '创建Workbench' and contains the following form fields:

- * 作业名: `Workbench-20240322-114649`
- * 账户: `a_nic`
- * 分区: `cpuHygon7380`
- * QOS: `normal`
- * 节点数: `1`
- * 单节点CPU核心数: `3`
- * 最长运行时间: `60` 分钟
- 其他sbatch参数: `比如: --gpu gres:2 --time 10`

At the bottom, it displays resource allocation: '总CPU核心数: 3' and '总内存容量: 11.8 GB'. There are '取消' and '提交' buttons at the bottom.



交互式应用

Browser address bar: <https://hpckapok.scut.edu.cn/a> 110%

Navigation: SCOW SUPER COMPUTING ON WEB | 仪表盘 | 作业 | Shell | 桌面 | **交互式应用** | 文件管理 | 目管理系统 | CN 简体中文

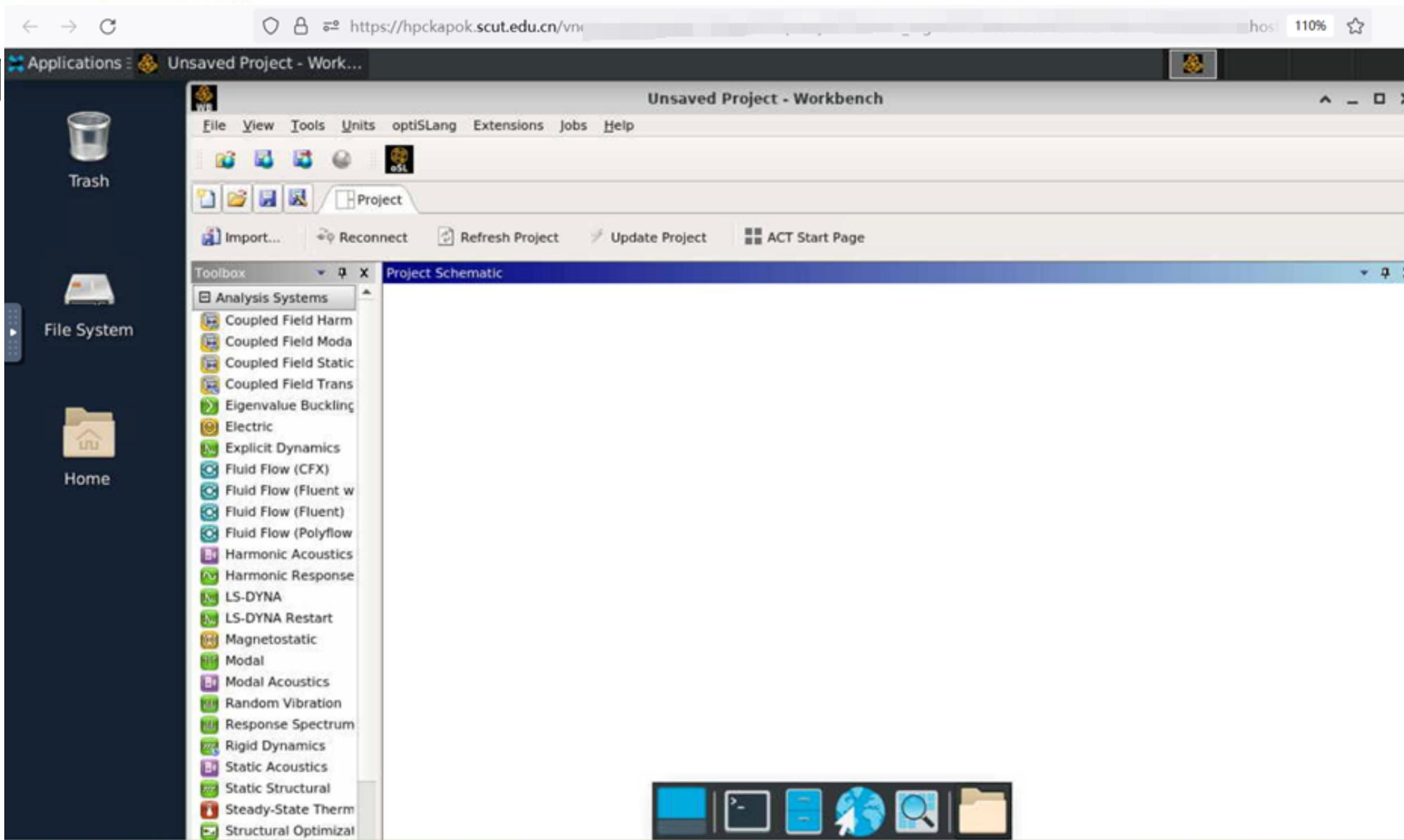
集群hpckapok2交互式应用

作业名: **搜索** **刷新** 10s自动刷新 只展示未结束的作业

作业名	作业ID	应用	提交时间	状态	剩余时间	操作
Workbench-20240322-124038	5708	Workbench	2024-03-22 12:40:42	RUNNING	59:40	连接 结束 进入目录



交互式应用





远程桌面(VNC)

Browser address bar: <https://hpckapok.scut.edu.cn/>

Navigation menu: SCOW SUPER COMPUTING ON WEB | 仪表盘 | 作业 | Shell | **桌面** | 交互式应用 | 文件管理 | 管理系统 | CN 简体中文

登录节点上的桌面

集群: hpckapok2 | 登录节点: [] | 刷新 | + 新建桌面

桌面ID	桌面名称	桌面类型	地址	创建时间	操作
2	desktop-20240310-212104	Xfce	login1	2024-03-10T21:21:09	启动 删除
1	desktop-20240313-094855	Xfce	admin	2024-03-13T09:49:01	启动 删除



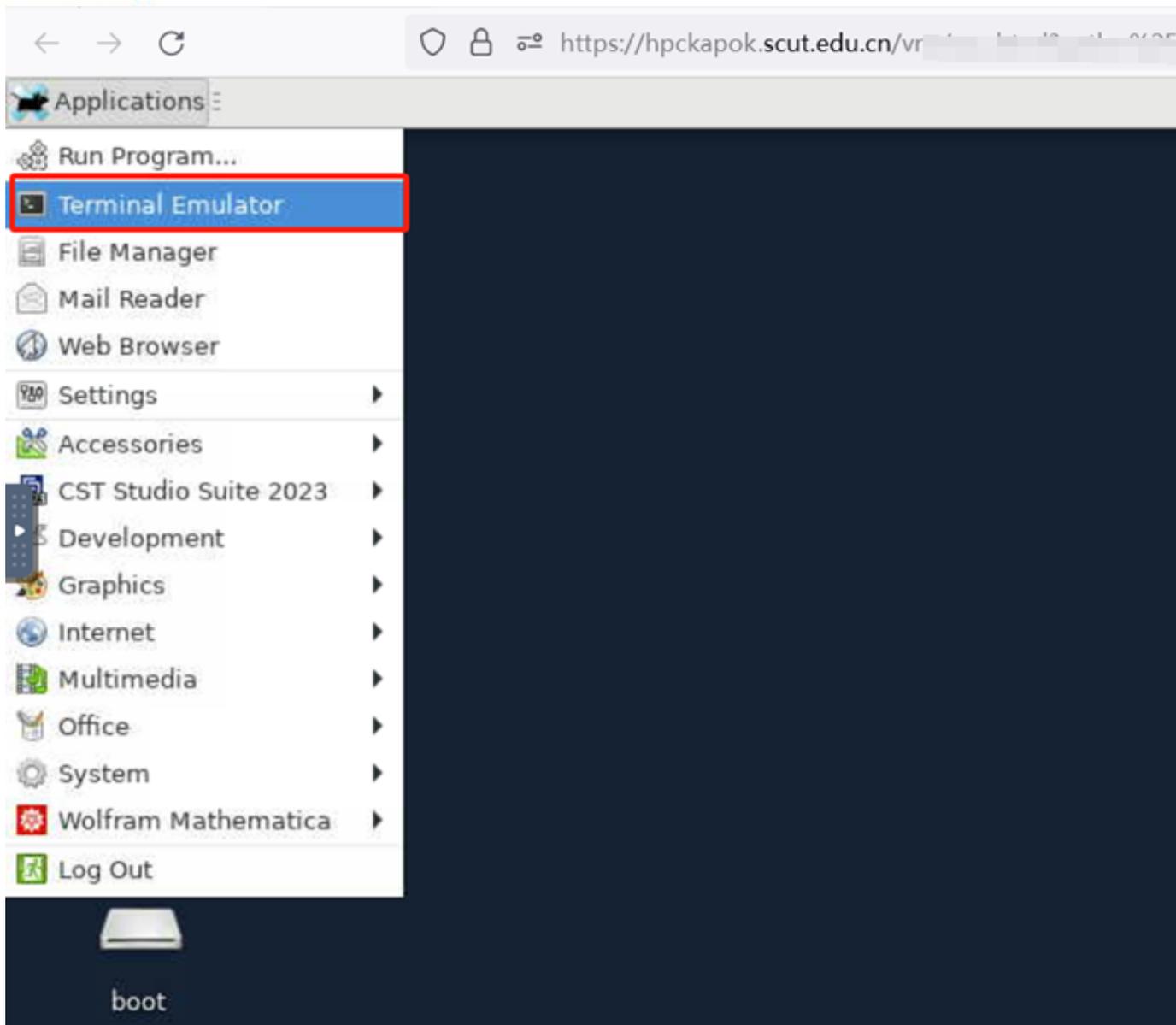
华南理工大学

South China University of Technology

科学计算公共服务平台

可视化方式

远程桌面(VNC)





- 远程桌面(VNC)

```
Applications: Xfce Terminal
Terminal - zqliu@comput2:~
File Edit View Terminal Tabs Help
[zqliu@admin root]$ salloc -A a_nicoper -N 1 -p cpuXeon6458 -t 60:00 --cpus-per-task=2
salloc: Pending job allocation 5707
salloc: job 5707 queued and waiting for resources
salloc: job 5707 has been allocated resources
salloc: Granted job allocation 5707
salloc: Waiting for resource configuration
salloc: Nodes comput2 are ready for job
[zqliu@admin root]$ ssh -Y comput2
activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Mar 22 12:00:49 2024 from 192.168.5.254
[zqliu@comput2 ~]$ export XDG_RUNTIME_DIR="$(mktemp -d)"
[zqliu@comput2 ~]$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/public/software/share/lib64/
[zqliu@comput2 ~]$ export PATH=/public/software/ansys_app/ansys_inc/v232/fluent/bin:/public/software/ansys_app/ansys_inc/v232/Framework/bin:/Linux64:$PATH
[zqliu@comput2 ~]$ export ANSYSLMD_LICENSE_FILE=1055@192.168.4.185
[zqliu@comput2 ~]$ runwb2
```



- 远程桌面(VNC)

```
$ salloc -N 1 -p cpuXeon6458 -t 60:00 --cpus-per-task=2
```

```
#申请计算节点
```

```
$ ssh -Y nodename
```

```
#登录计算节点
```

```
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/public/software/share/lib64/
```

```
$ export
```

```
PATH=/public/software/ansys_app/ansys_inc/v232/fluent/bin:/public/software/ansys_app/ansys_inc/v232/  
/Framework/bin/Linux64:$PATH
```

```
$ export ANSYSLMD_LICENSE_FILE=1055@192.168.4.111
```

```
#设置库文件，环境变量和license文件位置
```

```
$ runwb2
```

```
#运行workbench
```




华南理工大学

South China University of Technology

科学计算公共服务平台

六 账户管理



快捷入口



提交作业



未结束的作业

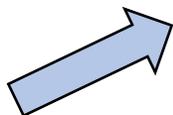


所有作业



作业模板

管理系统



未结束的作业

已结束的作业

集群和分区信息

操作日志

本用户已结束的作业

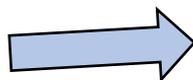
批量搜索 精确搜索

集群: hpckapok2 × hpckapok1 × 账户: 请选择账户 作业结束时间: 2

作业数量: 0 | 作业计费合计: 0.000 元

集群作业ID	作业名	账户	集
--------	-----	----	---

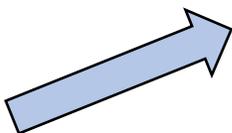
成员账号界面





The screenshot shows the '账户管理' (Account Management) section of the SCOW platform. The main area displays '账户a_...扣费记录' (Account a_... Charge Record). It includes search filters for '用户' (User), '时间' (Time), and '类型' (Type), along with a summary: '总数: 8 合计: 0.145'. Below the filters is a table of charges:

用户	扣费日期	扣费金额	类型
[blurred]	2024-04-23 15:10:14	0.101	作业费用
[blurred]	2024-04-23 10:20:13	0.001	作业费用
[blurred]	2024-04-22 17:20:13	0.003	作业费用



负责人账号界面

作业管理：过往作业，操作日志，消费记录等。

用户管理：账户所有者/负责人可以移除名下用户，可设置管理员，进行限额。

账户状态：1.正常 2.封锁/暂停：该状态下用户可以登录但不可以提交作业；用户连续两个月未缴纳账单将被封锁；用户可申请封锁/暂停。3.停止：该状态下用户不可以登录 4.销户：该状态下用户数据将会被删除；账户注销前，需在缴清该账户的欠费后方可提交注销申请；用户离校、离职或不再使用，须主动申请注销账户或账号。



- 后付费。每个月系统产生一条账单，推送至学校网上办事大厅，并邮件通知。
- 登录“网上办事大厅 (<https://ehall.scut.edu.cn>) ” ---> “校内转账缴科学计算费” 缴纳费用

校内转账缴科学计算费							
姓名	刘志权			所在单位	信息网络工程研究中心		
手机号码	<input type="text" value="13632198755"/>			缴费主账号	<input type="text" value="a_nic"/>		
账户封锁状态	否						
缴费月份	<input type="text" value="202402,"/>						
详细账单	202402: {"作业费用": "0.01", "基本账户费": "50"};						
缴费金额	50.01						
账单详情	<input type="button" value="导出数据"/>						
	序号	人事编号/学号	姓名	单位名称	月份	金额	详情
	1	zqliu	刘志权		202402	50	{"基本账户费": "50"}
2	guozuoen	郭祚恩		202402	0.01	{"作业费用": "0.01"}	
经费卡号:	<input type="text" value="请选择"/>						
部门编号:				项目编号:			
员工编号:				国库信息码			
经费校验:	<input type="text" value="经费项目校验"/>						
校验结果							



华南理工大学
South China University of Technology

科学计算公共服务平台

七 常见问题



华南理工大学
South China University of Technology

科学计算公共服务平台

Q:如何查看作业的运行情况?

A:可以查看日志, 提交作业的时候可以添加`--output=%j.out`参数输出文件, `--error=%j.err`错误文件, 可以用`tail -f %j.out`查看作业是否有输出或者报错, %j为作业的id。也可以直接登录申请到的计算节点查看cpu利用率。

Q:为什么我在平台上跑的比我电脑还慢?

A: 出现这种原因可能是代码没有使用好cpu或者gpu, 比如多线程或者并行机制没有发挥作用, 首先可以登录到计算节点查看你 cpu利用率, 然后查看代码是否要相关参数来调整, 比如高斯要设置`%nprocshared`, python可能是设`multiprocessing`, 最后选择一个有效资源(请查看https://hpc.scut.edu.cn/docs/job/gres_select.html)。



华南理工大学
South China University of Technology

科学计算公共服务平台

Q:计算节点不能访问互联网/不能下载数据?

A:目前计算节点禁止连接外部网络, 只有登录节点可以访问外网。可以在登录节点下载数据。

Q:网页上无法上传超过1G的文件, 大文件如何上传?

A:可以用winscp或mobaxterm等软件直接上传文件。

Q:单个作业运行时间多久, 如何设置运行时间?

A:目前没有对作业时间进行限制, 后续有时间限制, 会提示。设置时间可以在提交作业时候添加-t 1-2:30:30。或者在脚本中添加#SBATCH -t 1-2:30:30。



华南理工大学
South China University of Technology

科学计算公共服务平台

Q:平台显示无法找到mpifort或者库文件?

A:在当前路径找不到mpifort编译器或者库文件，可能是没有加载编译环境，可以使用module av查看系统中有哪些编译器可以使用的，库文件也可以在目录下用find命令查找，然后使用export命令来加载。

Q:我提交任务后，显示提交成功，但是运行一天后发现没有输出结果，是集群出问题了吗?

A:建议提交作业后，及时查看作业输出日志文件，确保作业正常运行。若作业异常请及时取消作业避免产生不必要的费用。当出现作业不正常运行情况，需要具体分析作业脚本和运行程序，如需协助，请保持作业目录现状联系我们检查分析。



Q:如何重置 .bashrc 和 .bash_profile ?

A:用户目录下的 ~/.bashrc 和 ~/.bash_profile 记录bash shell配置，若配置不当可能会导致无法找到可执行文件等问题，需要重置这两个配置文件的内容。建议您先备份这两个文件，然后执行如下命令重置：

```
$ cp /etc/skel/.bash* ~
```

```
$ source ~/.bashrc
```

重置配置文件会导致您先前对bash shell的自定义配置失效，如果您仍需要保留这些自定义配置，建议利用备份文件逐条转移这些配置，避免引入导致应用异常语句。

Q:请问提交任务出现这样的错误是什么原因呢
sbatch: error: Batch job submission failed: Invalid account or account/partition combination specified

A:出现此原因可能是账户欠费了，可以在主门户查看账户状态是否为“封锁”。



华南理工大学

South China University of Technology

科学计算公共服务平台

技术支持

电话: 020-87114020

网站: <https://hpc.scut.edu.cn>

电子邮件: hpc@scut.edu.cn

QQ群: 263814144 (华工算力服务)

BBS: <https://bbs.scut.edu.cn> HPC+AI (学术天地)