联系电话：021-64032612
联系邮箱：hpc@ion.ac.cn
联系地址：岳阳路319号8号楼605

CEBSIT CPU集群使用培训

脑科学数据与计算中心
2021年9月17日

一 使用集群计算资源的方式

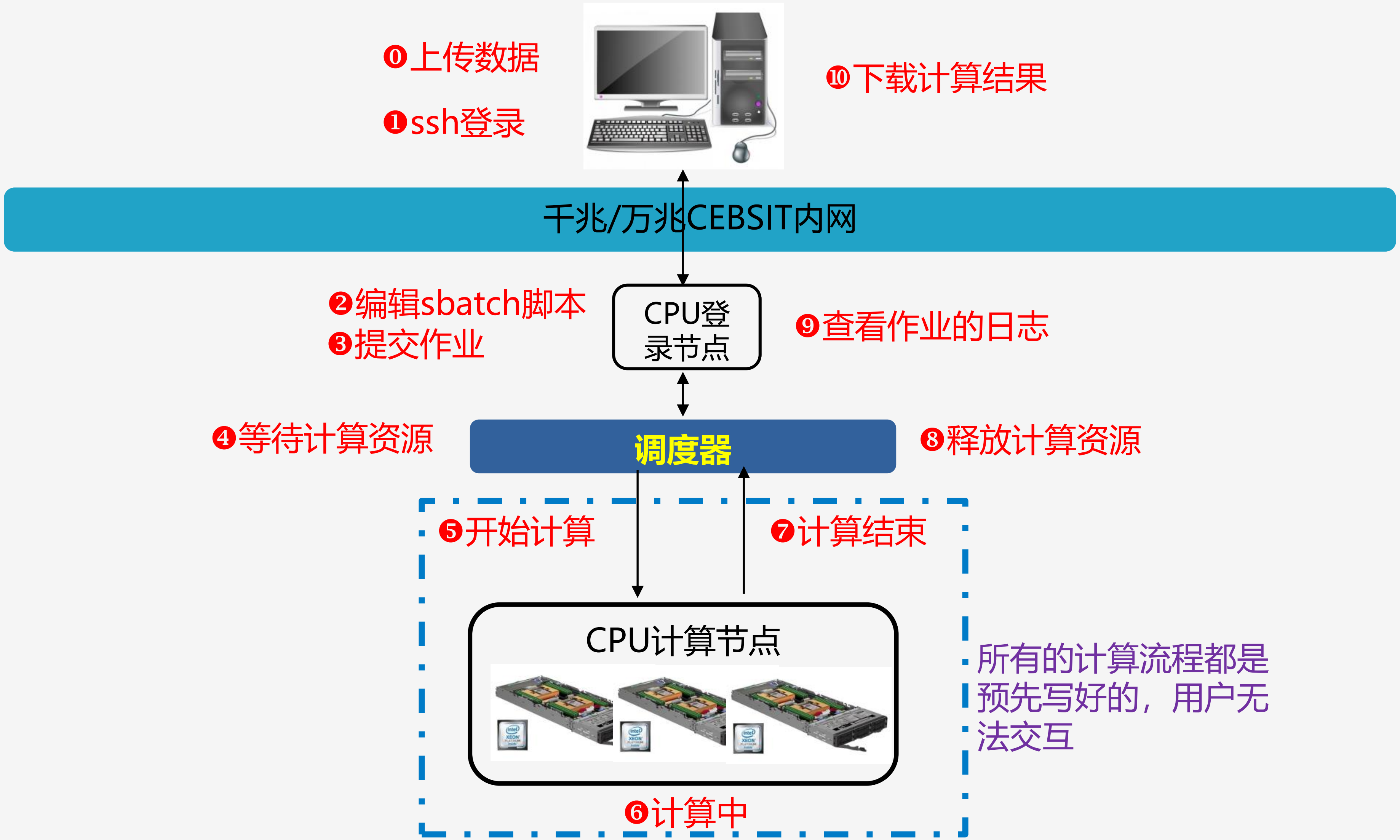
二 集群硬件和软件栈介绍

三 CPU集群容器工具介绍与容器示例

四 交互式使用计算资源的详细步骤

五 集群使用注意事项和账户申请流程

最一般的使用集群计算的流程



⓿上传数据

❶ssh登录

❿下载计算结果

千兆/万兆CEBSIT内网

❷编辑sbatch脚本

❸提交作业

CPU登录节点

❾查看作业的日志

❹等待计算资源

调度器

❽释放计算资源

❺开始计算

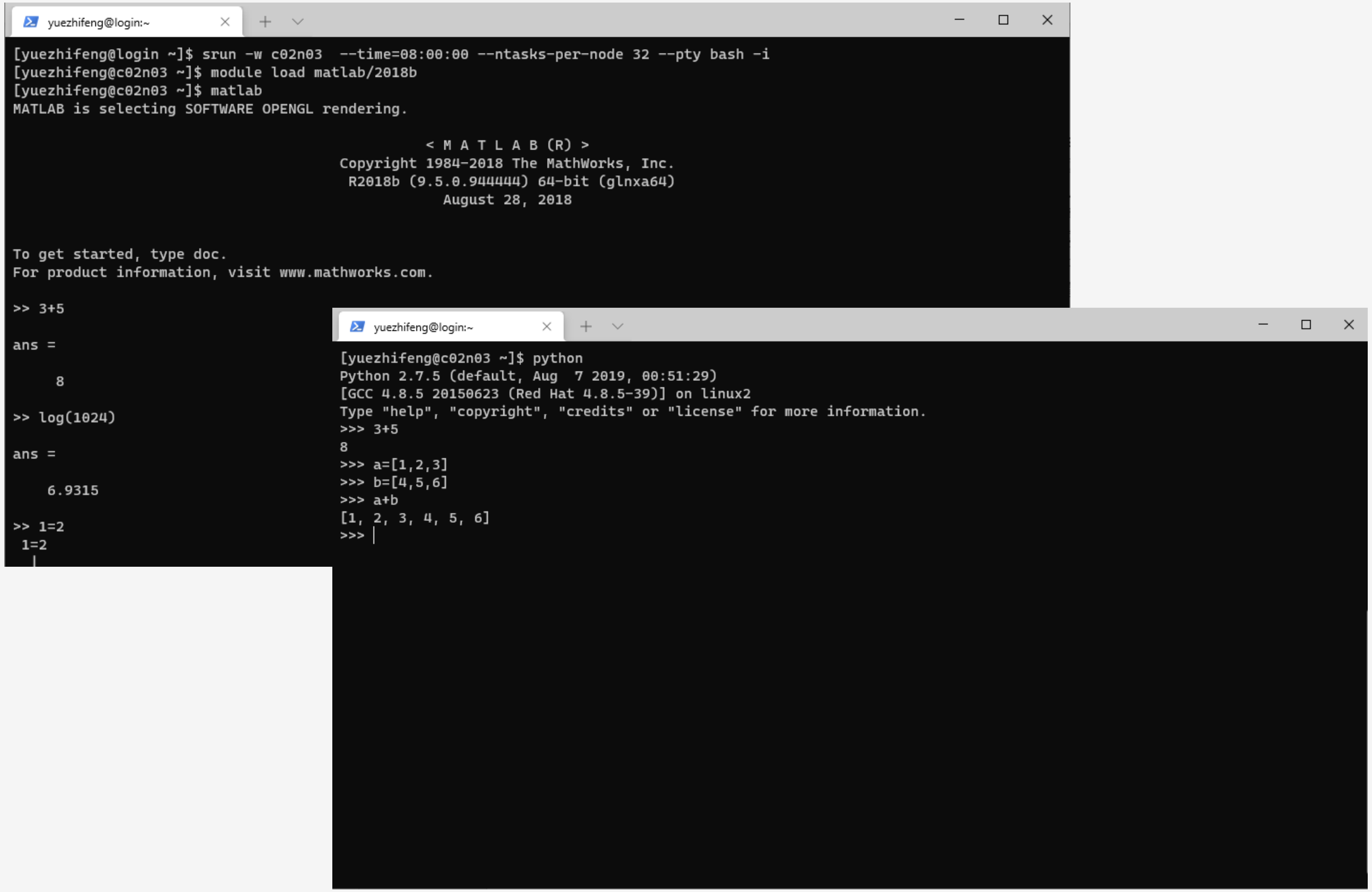
❼计算结束

CPU计算节点

❻计算中

所有的计算流程都是预先写好的，用户无法交互

交互式使用计算节点的方式：1. 命令行模式



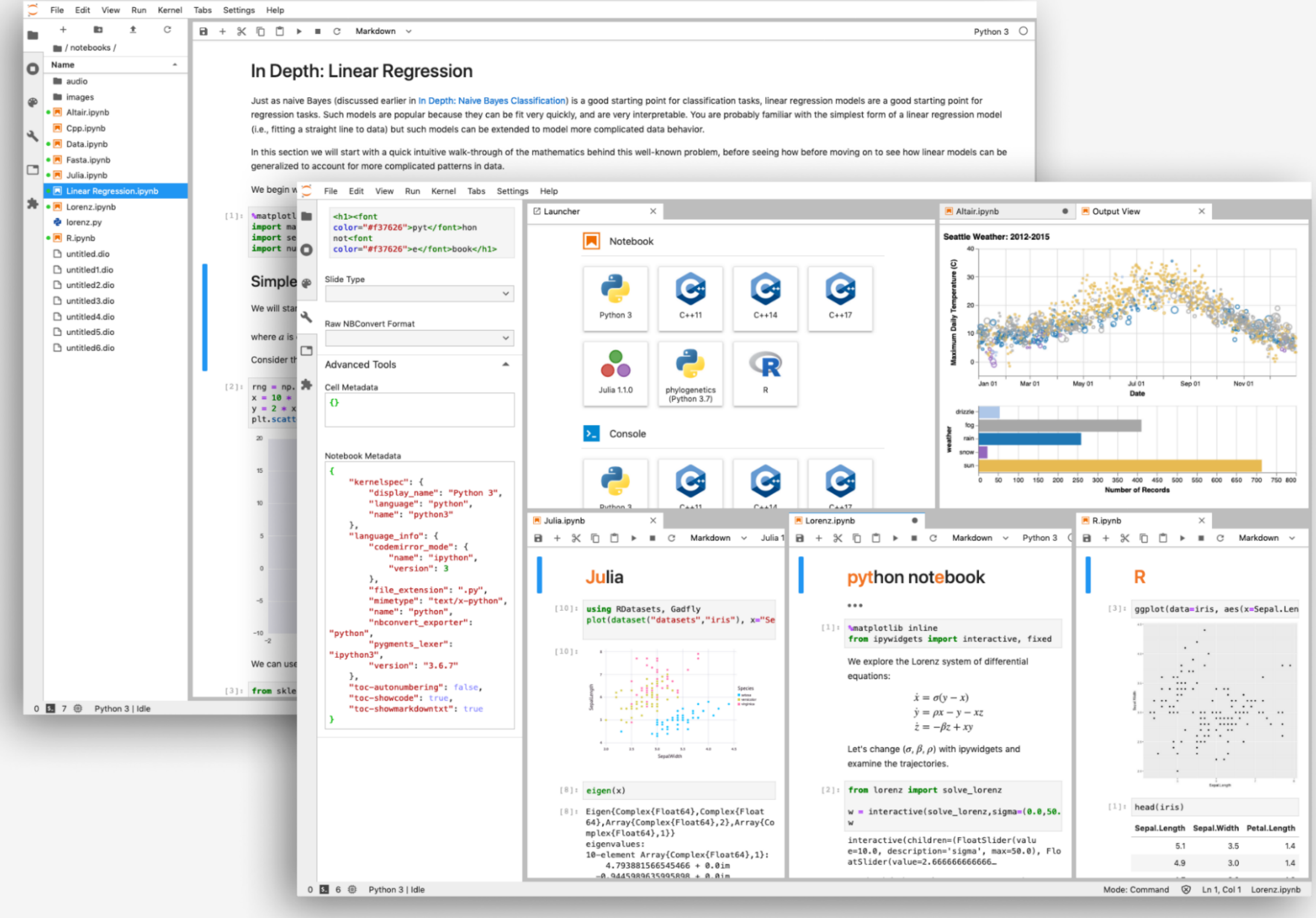
```
[yuezhifeng@login ~]$ srun -w c02n03  --time=08:00:00 --ntasks-per-node 32 --pty bash -i
[yuezhifeng@c02n03 ~]$ module load matlab/2018b
[yuezhifeng@c02n03 ~]$ matlab
MATLAB is selecting SOFTWARE OPENGL rendering.

                                      < M A T L A B (R) >
                            Copyright 1984-2018 The MathWorks, Inc.
                             R2018b (9.5.0.944444) 64-bit (glnxa64)
                                        August 28, 2018


To get started, type doc.
For product information, visit www.mathworks.com.

>> 3+5

ans =

     8

>> log(1024)

ans =

    6.9315

>> 1=2
 1=2
  |
```

```
[yuezhifeng@c02n03 ~]$ python
Python 2.7.5 (default, Aug  7 2019, 00:51:29)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-39)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 3+5
8
>>> a=[1,2,3]
>>> b=[4,5,6]
>>> a+b
[1, 2, 3, 4, 5, 6]
>>> |
```

为了充分使用计算机提供的计算资源，早期很多计算机会连接若干终端控制台，这些终端控制台从硬件上构造很简单，只包括键盘和显示器，**不执行计算的任务，只简单的把用户的输入发送到主计算机去处理，然后再把计算结果返回给用户**。从软件使用上看，只提供给用户一个使用命令行的字符界面，用于接收用户输入和反馈计算结果。

像Windows下的命令行状态，Linux、Unix下的字符终端程序，这些就称为虚拟控制台。

交互式使用计算节点的方式：2. Jupyter



In Depth: Linear Regression

Just as naive Bayes (discussed earlier in In Depth: Naive Bayes Classification) is a good starting point for classification tasks, linear regression models are a good starting point for regression tasks. Such models are popular because they can be fit very quickly, and are very interpretable. You are probably familiar with the simplest form of a linear regression model (i.e., fitting a straight line to data) but such models can be extended to model more complicated data behavior.

In this section we will start with a quick intuitive walk-through of the mathematics behind this well-known problem, before seeing how before moving on to see how linear models can be generalized to account for more complicated patterns in data.

```
<h1><font
color="#f37626">pyt</font>hon
not<font
color="#f37626">e</font>book</h1>
```

Slide Type

Raw NBConvert Format

Advanced Tools

Cell Metadata

```
{}
```

Notebook Metadata

```
{
    "kernelspec": {
        "display_name": "Python 3",
        "language": "python",
        "name": "python3"
    },
    "language_info": {
        "codemirror_mode": {
            "name": "ipython",
            "version": 3
        },
        "file_extension": ".py",
        "mimetype": "text/x-python",
        "name": "python",
        "nbconvert_exporter":
"python",
        "pygments_lexer":
"ipython3",
        "version": "3.6.7"
    },
    "toc-autonumbering": false,
    "toc-showcode": true,
    "toc-showmarkdowntxt": true
}
```

Julia

```
[10]: using RDatasets, Gadfly
      plot(dataset("datasets","iris"), x="Se
```

```
[8]: eigen(x)
[8]: Eigen{Complex{Float64},Complex{Float
64},Array{Complex{Float64},2},Array{Co
mplex{Float64},1}}
eigenvalues:
10-element Array{Complex{Float64},1}:
    4.793881566545466 + 0.0im
```

python notebook

```
[1]: %matplotlib inline
     from ipywidgets import interactive, fixed
```

We explore the Lorenz system of differential equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

Let's change (σ, β, ρ) with ipywidgets and examine the trajectories.

```
[2]: from lorenz import solve_lorenz

     w = interactive(solve_lorenz,sigma=(0.0,50.
     w
interactive(children=(FloatSlider(valu
e=10.0, description='sigma', max=50.0), Flo
atSlider(value=2.666666666666…
```

R

```
[3]: ggplot(data=iris, aes(x=Sepal.Len
```

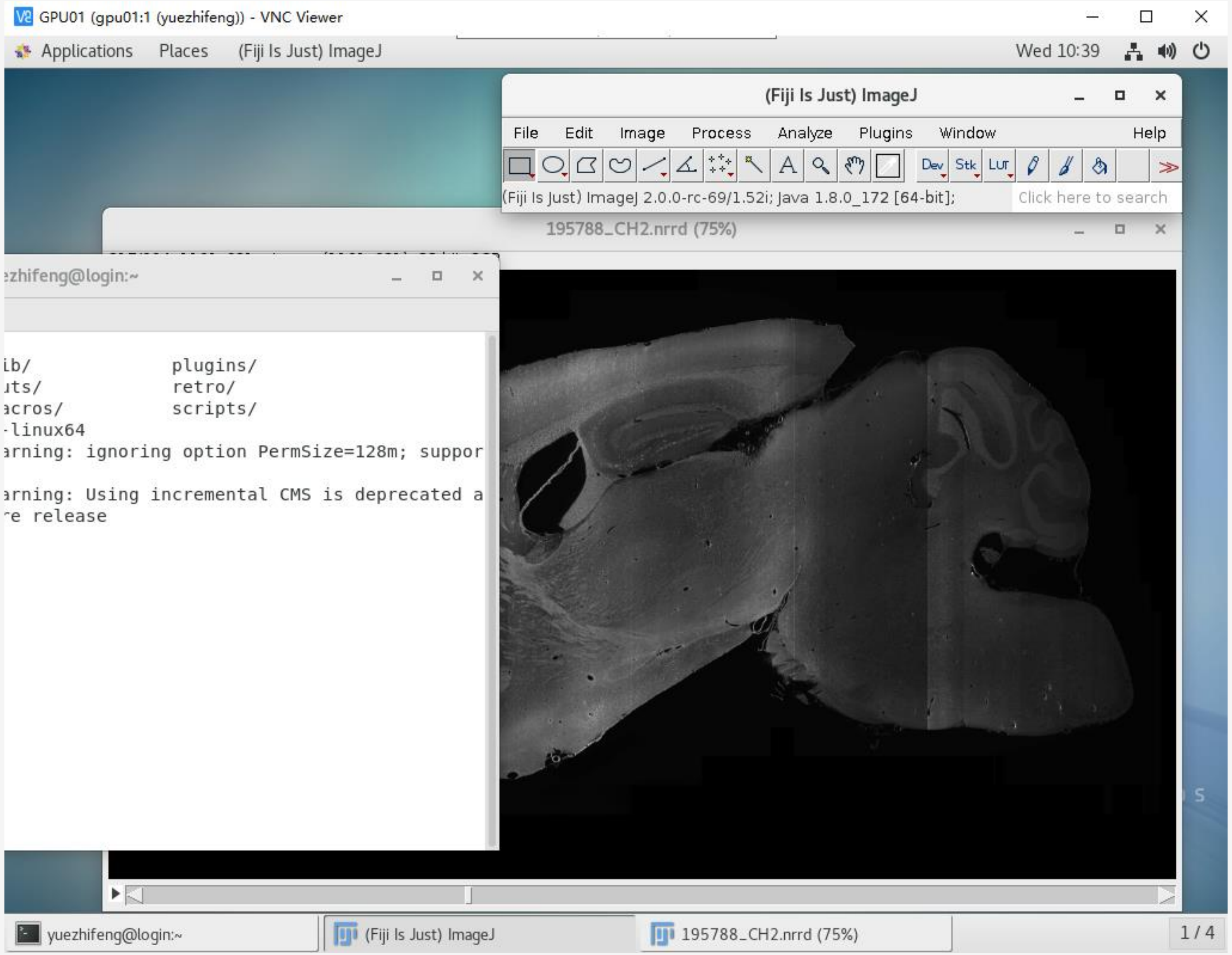
```
[1]: head(iris)
```

Sepal.Length	Sepal.Width	Petal.Length
5.1	3.5	1.4
4.9	3.0	1.4

Jupyter Notebook是基于网页的用于交互计算的应用程序。其可被应用于全过程计算：开发、文档编写、运行代码和展示结果。——Jupyter Notebook官方介绍

简而言之，Jupyter Notebook是以网页的形式打开，可以在网页页面中**直接编写代码**和**运行代码**，代码的**运行结果**也会直接在代码块下显示的程序。如在编程过程中需要编写说明文档，可在同一个页面中直接编写，便于作及时的说明和解释。

交互式使用计算节点的方式：3. VNC



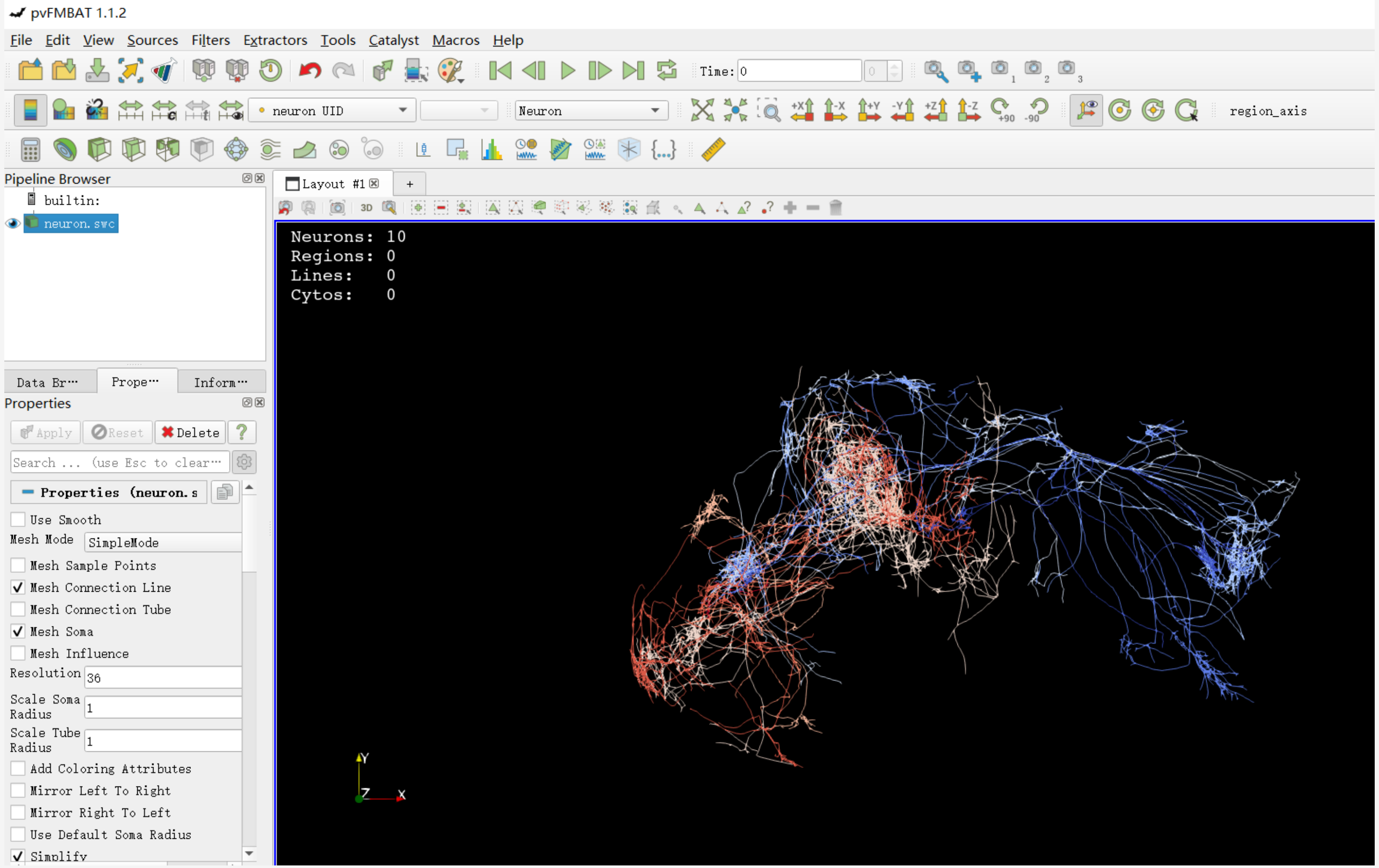
(Fiji Is Just) ImageJ 2.0.0-rc-69/1.52i; Java 1.8.0_172 [64-bit];

```
ib/             plugins/
uts/            retro/
acros/          scripts/
-linux64
arning: ignoring option PermSize=128m; suppor

arning: Using incremental CMS is deprecated a
re release
```

VNC (Virtual Network Console)，即虚拟网络控制台，它是一款基于 UNIX 和 Linux 操作系统的优秀远程控制工具软件，由著名的 AT&T 的欧洲研究实验室开发，远程控制能力强大，高效实用，并且免费开源。

VNC基本上是由两部分组成：一部分是客户端的应用程序(vncviewer)；另外一部分是服务器端的应用程序(vncserver)。在任何安装了客户端的应用程序(vncviewer)的计算机都能十分方便地与安装了服务器端的应用程序(vncserver)的计算机相互连接。

交互式使用计算节点的方式：4. pvFMbat



```
Neurons:  10
Regions:  0
Lines:    0
Cytos:    0
```

Properties (neuron.s

- ☐ Use Smooth
- Mesh Mode: SimpleMode
- ☐ Mesh Sample Points
- ☑ Mesh Connection Line
- ☐ Mesh Connection Tube
- ☑ Mesh Soma
- ☐ Mesh Influence
- Resolution: 36
- Scale Soma Radius: 1
- Scale Tube Radius: 1
- ☐ Add Coloring Attributes
- ☐ Mirror Left To Right
- ☐ Mirror Right To Left
- ☐ Use Default Soma Radius
- ☑ Simplify

pvFMbat是脑科学数据与计算中心与平台和研究组联合研发的交互式脑图谱数据可视化和分析平台。pvFMbat基于开源软件paraview，我们自研的C++代码量有几万行规模。pvFMbat以client-sever模式运行。Client安装运行在个人电脑，server安装运行在CEBSIT集群的gpu01节点。

Server端的GPU资源可以用来加速渲染，更快可视化神经元、脑区、各种3DImage。

一 使用集群计算资源的方式

二 **集群硬件和软件栈介绍**

三 CPU集群容器工具介绍与容器示例

四 交互式使用计算资源的详细步骤

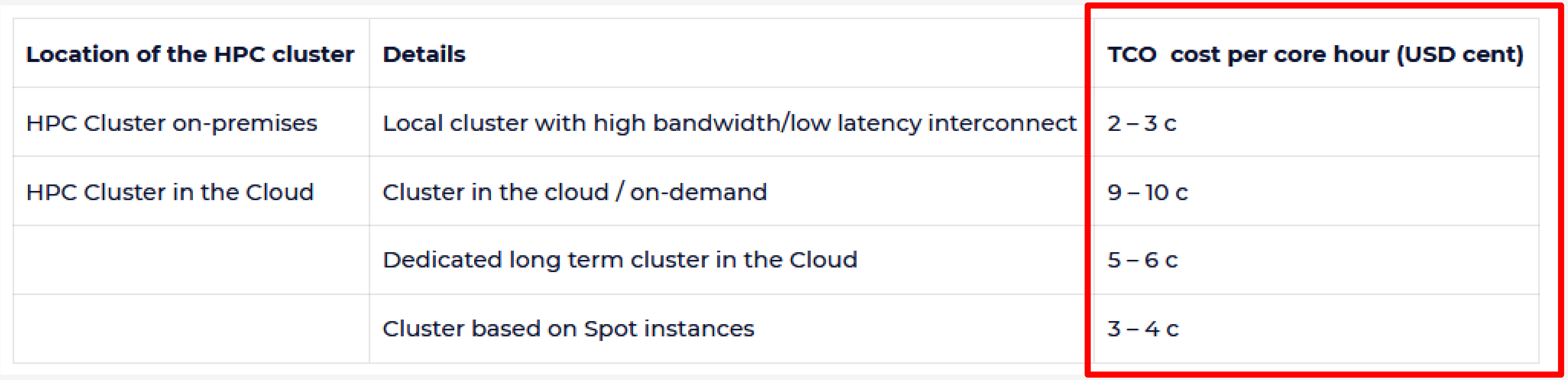
五 集群使用注意事项和账户申请流程

CEBSIT CPU 集群硬件总览

队列	节点数	单节点资源
low	16	32核；256GB内存；
earth	12	48核；768GB内存；1.5TB SSD；
fat01	1	96核；3TB内存；
fat02	1	96核；3TB内存；
fat03	1	224核；6TB内存；
gpu	1	32核；512GB内存；4 x V100 (16GB)
总数	32	1536核；25.5TB内存；

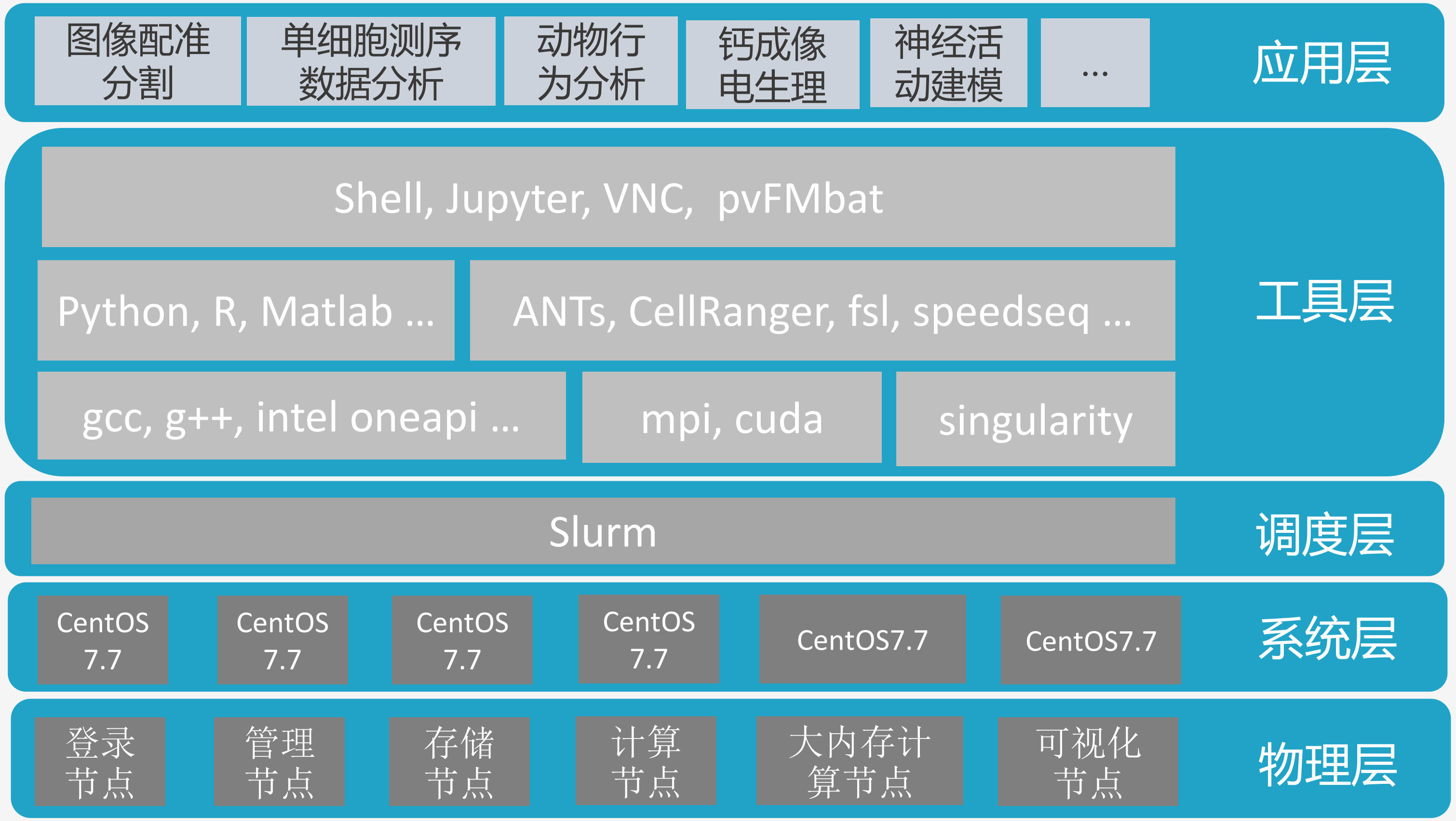
HPC Price On-prem vs Cloud

https://www.ni-sp.com/support/hpc-in-the-cloud



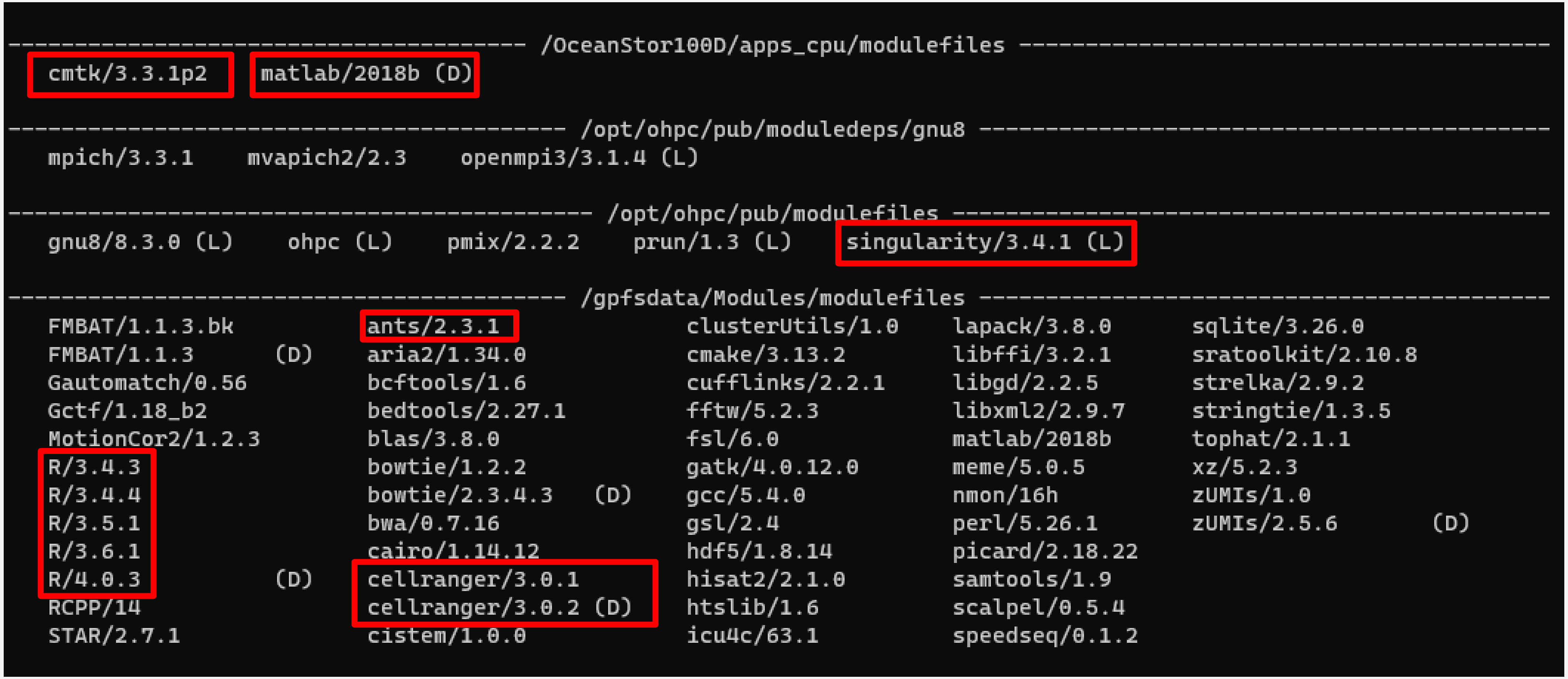
Location of the HPC cluster	Details	TCO cost per core hour (USD cent)
HPC Cluster on-premises	Local cluster with high bandwidth/low latency interconnect	2 – 3 c
HPC Cluster in the Cloud	Cluster in the cloud / on-demand	9 – 10 c
	Dedicated long term cluster in the Cloud	5 – 6 c
	Cluster based on Spot instances	3 – 4 c

CEBSIT CPU 软件栈简介



CEBSIT CPU 集群软件栈

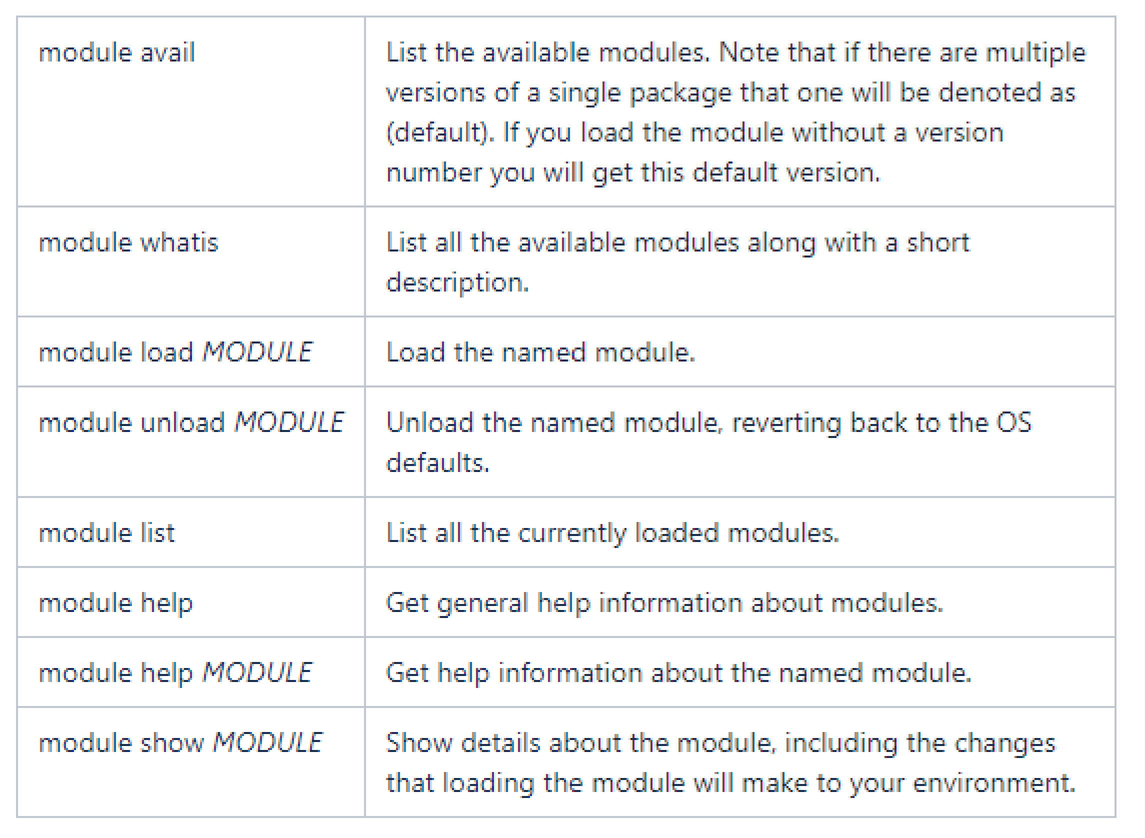
CEBSIT CPU 工具软件列表



```
---------------------------------------- /OceanStor100D/apps_cpu/modulefiles ----------------------------------------
   cmtk/3.3.1p2     matlab/2018b (D)

------------------------------------------ /opt/ohpc/pub/moduledeps/gnu8 -------------------------------------------
   mpich/3.3.1    mvapich2/2.3    openmpi3/3.1.4 (L)

-------------------------------------------- /opt/ohpc/pub/modulefiles --------------------------------------------
   gnu8/8.3.0 (L)    ohpc (L)    pmix/2.2.2    prun/1.3 (L)    singularity/3.4.1 (L)

------------------------------------------ /gpfsdata/Modules/modulefiles -------------------------------------------
   FMBAT/1.1.3.bk          ants/2.3.1              clusterUtils/1.0    lapack/3.8.0       sqlite/3.26.0
   FMBAT/1.1.3      (D)    aria2/1.34.0            cmake/3.13.2        libffi/3.2.1       sratoolkit/2.10.8
   Gautomatch/0.56         bcftools/1.6            cufflinks/2.2.1     libgd/2.2.5        strelka/2.9.2
   Gctf/1.18_b2            bedtools/2.27.1         fftw/5.2.3          libxml2/2.9.7      stringtie/1.3.5
   MotionCor2/1.2.3        blas/3.8.0              fsl/6.0             matlab/2018b       tophat/2.1.1
   R/3.4.3                 bowtie/1.2.2            gatk/4.0.12.0       meme/5.0.5         xz/5.2.3
   R/3.4.4                 bowtie/2.3.4.3   (D)    gcc/5.4.0           nmon/16h           zUMIs/1.0
   R/3.5.1                 bwa/0.7.16              gsl/2.4             perl/5.26.1        zUMIs/2.5.6       (D)
   R/3.6.1                 cairo/1.14.12           hdf5/1.8.14         picard/2.18.22
   R/4.0.3          (D)    cellranger/3.0.1        hisat2/2.1.0        samtools/1.9
   RCPP/14                 cellranger/3.0.2 (D)    htslib/1.6          scalpel/0.5.4
   STAR/2.7.1              cistem/1.0.0            icu4c/63.1          speedseq/0.1.2
```

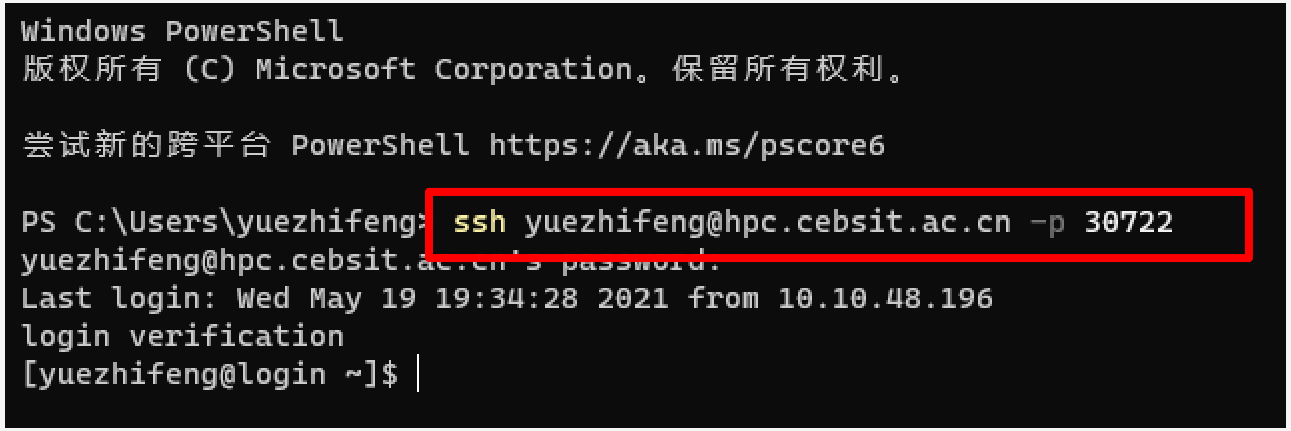
module：用来管理软件环境的工具

https://hpc.pku.edu.cn/_book/guide/soft_env/module.html
https://docs.hpc.sjtu.edu.cn/app/module.html?highlight=module#module



module avail	List the available modules. Note that if there are multiple versions of a single package that one will be denoted as (default). If you load the module without a version number you will get this default version.
module whatis	List all the available modules along with a short description.
module load *MODULE*	Load the named module.
module unload *MODULE*	Unload the named module, reverting back to the OS defaults.
module list	List all the currently loaded modules.
module help	Get general help information about modules.
module help *MODULE*	Get help information about the named module.
module show *MODULE*	Show details about the module, including the changes that loading the module will make to your environment.

命令行SSH登陆集群：以系统自带PowerShell为例



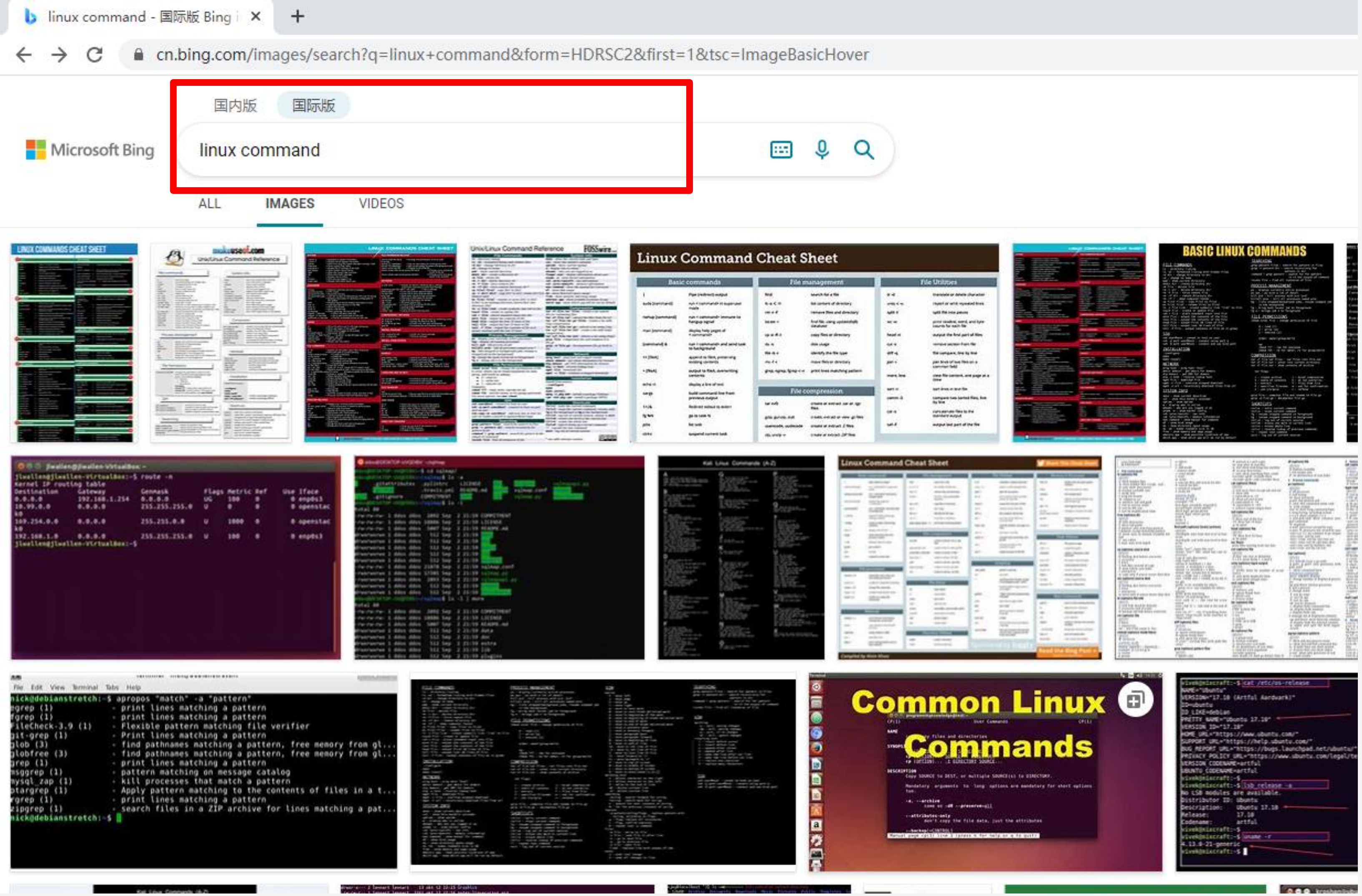
```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$ 
```

ssh yourname@hpc.cebsit.ac.cn –p 30722

Linux基础命令

1. ls : 列出当前或指定目录下的文件或目录
2. passwd: 更改用户登录密码
3. logout: 退出登录
4. pwd: 显示当前目录
5. cd : 进入指定目录
6. chmod :更改文件读、写或执行权限
7. rm: 删除文件或目录
8. cp: 拷贝文件或目录
9. find: 在指定目录下查找文件
10. mv : 文件更名或移动
11. vi :文本编辑器
12. top : 查看系统长时间运行的主要进程
13. ps -ef:查看系统进程。
14. kill : 杀掉一个指定进程号的进程或向系统发送一个信号。
15. man : 给出指定命令的详细描述。
16. date :显示或设置系统时间。
17. rcp,ftp, sftp,scp: 远程文件拷贝。

最快的学习方式是搜索“linux command”


linux command - 国际版 Bing
国内版
国际版
Microsoft Bing
ALL
IMAGES
VIDEOS
LINUX COMMANDS CHEAT SHEET
Unix/Linux Command Reference
Linux Command Cheat Sheet
Basic commands
File management
File Utilities
File compression
BASIC LINUX COMMANDS
Kali Linux Commands (A-Z)
Common Linux Commands
File Edit View Terminal Tabs Help

cn.bing.com/images/search?q=linux+command&form=HDRSC2&first=1&tsc=ImageBasicHover

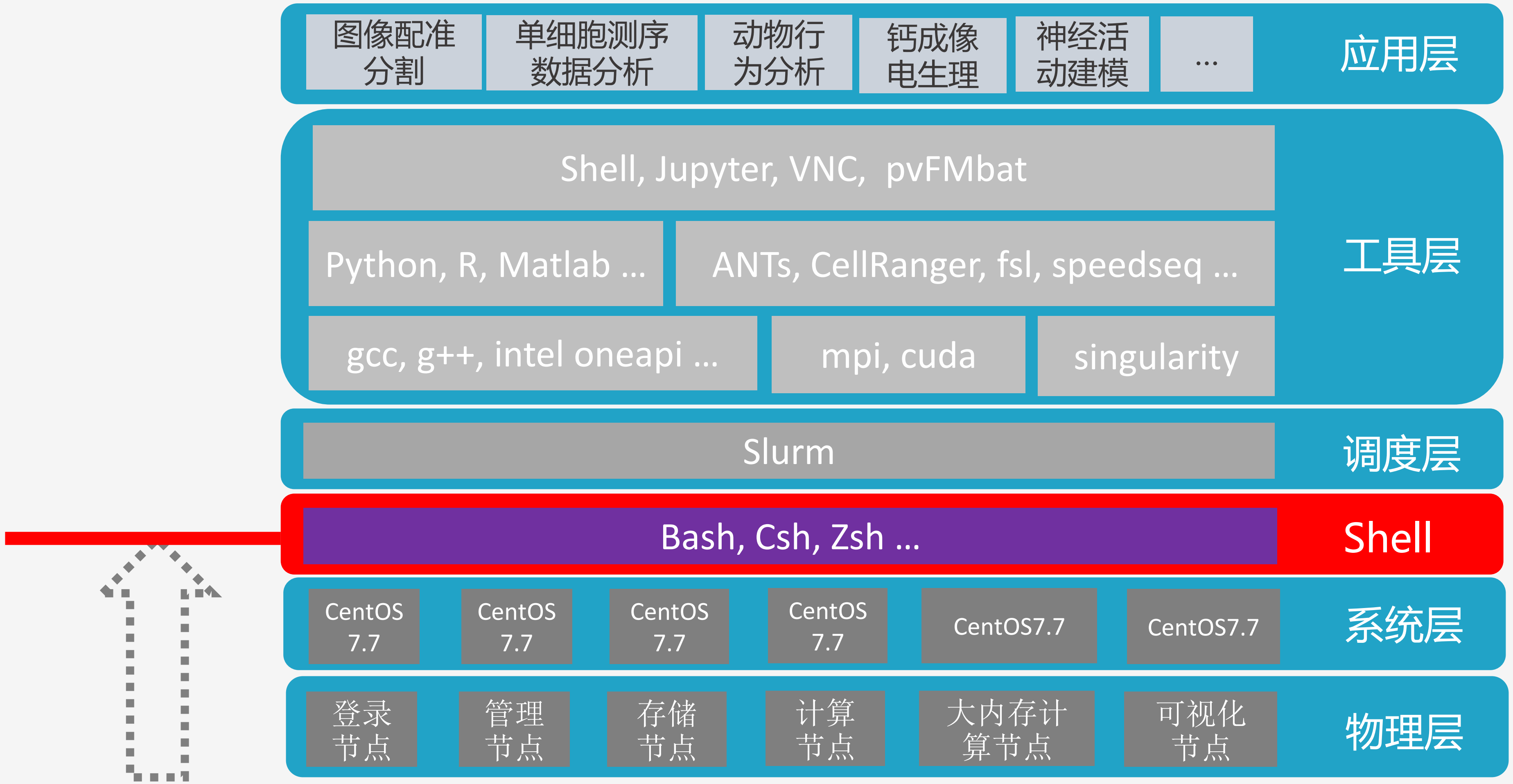
linux command

```
jlwallen@jlwallen-VirtualBox:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.1.254   0.0.0.0         UG    100    0        0 enp0s3
10.99.0.0       0.0.0.0         255.255.255.0   U     0      0        0 openstac
k0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 openstac
k0
192.168.1.0     0.0.0.0         255.255.255.0   U     100    0        0 enp0s3
jlwallen@jlwallen-VirtualBox:~$
```

```
nick@debianstretch:~$ apropos "match" -a "pattern"
egrep (1)            - print lines matching a pattern
fgrep (1)            - print lines matching a pattern
FileCheck-3.9 (1)    - Flexible pattern matching file verifier
git-grep (1)         - Print lines matching a pattern
glob (3)             - find pathnames matching a pattern, free memory from gl...
globfree (3)         - find pathnames matching a pattern, free memory from gl...
grep (1)             - print lines matching a pattern
msggrep (1)          - pattern matching on message catalog
mysql_zap (1)        - kill processes that match a pattern
ptargrep (1)         - Apply pattern matching to the contents of files in a t...
rgrep (1)            - print lines matching a pattern
zipgrep (1)          - search files in a ZIP archive for lines matching a pat...
nick@debianstretch:~$
```

```
vivek@nixcraft:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="17.10 (Artful Aardvark)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 17.10"
VERSION_ID="17.10"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/te
VERSION_CODENAME=artful
UBUNTU_CODENAME=artful
vivek@nixcraft:~$
vivek@nixcraft:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 17.10
Release:        17.10
Codename:       artful
vivek@nixcraft:~$
vivek@nixcraft:~$ uname -r
4.13.0-21-generic
vivek@nixcraft:~$
```

ssh登录后就可以指挥slurm了



应用层：图像配准分割 | 单细胞测序数据分析 | 动物行为分析 | 钙成像电生理 | 神经活动建模 | …

工具层：
Shell, Jupyter, VNC, pvFMbat
Python, R, Matlab … | ANTs, CellRanger, fsl, speedseq …
gcc, g++, intel oneapi … | mpi, cuda | singularity

调度层：Slurm

Shell：Bash, Csh, Zsh …

系统层：CentOS 7.7 | CentOS 7.7 | CentOS 7.7 | CentOS 7.7 | CentOS7.7 | CentOS7.7

物理层：登录节点 | 管理节点 | 存储节点 | 计算节点 | 大内存计算节点 | 可视化节点

ssh登录后就来
到了这个位置了

CEBSIT HPC 集群软件栈

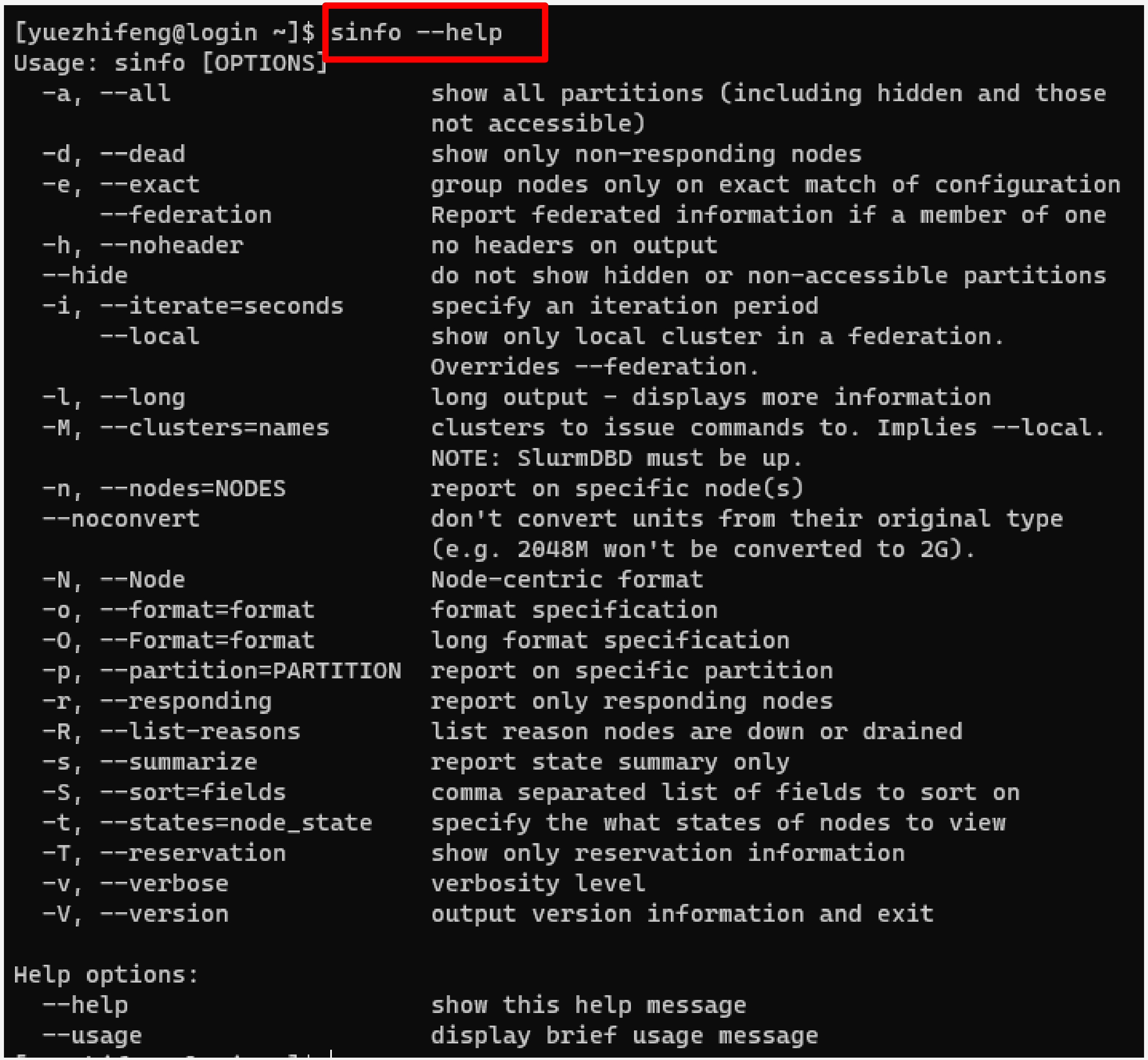
Slurm常用命令

命令	功能
sacct	查看历史作业信息
salloc	分配资源
sbatch	提交批处理作业
scancel	取消作业
scontrol	系统与作业控制
sinfo	查看节点与分区状态
squeue	查看队列状态
srun	执行作业

sinfo

- 学习sinfo最好的方法就是看help

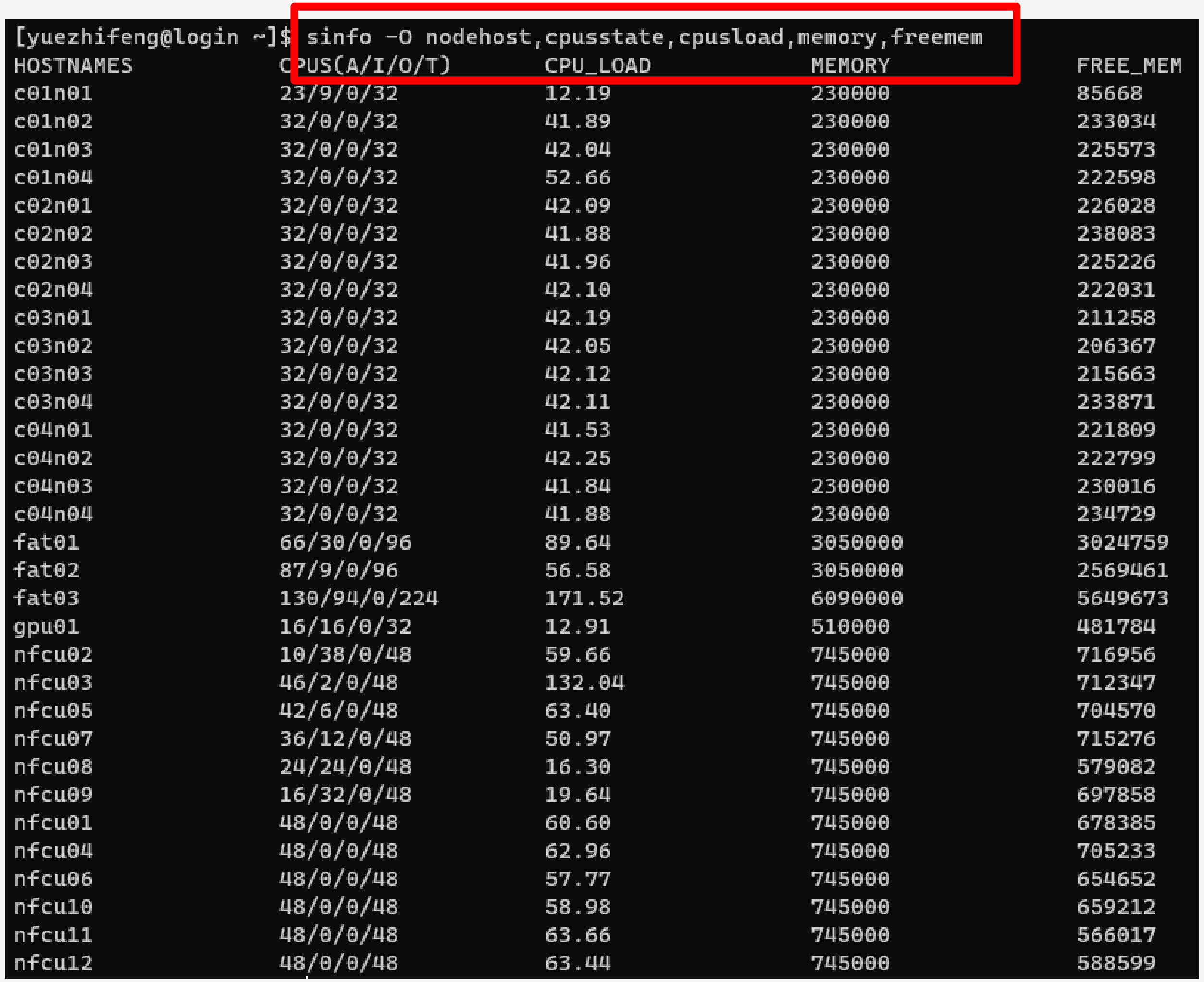
- https://slurm.schedmd.com/sinfo.html
- http://hpc.pku.edu.cn/_book/guide/slurm/sinfo.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#sinfo



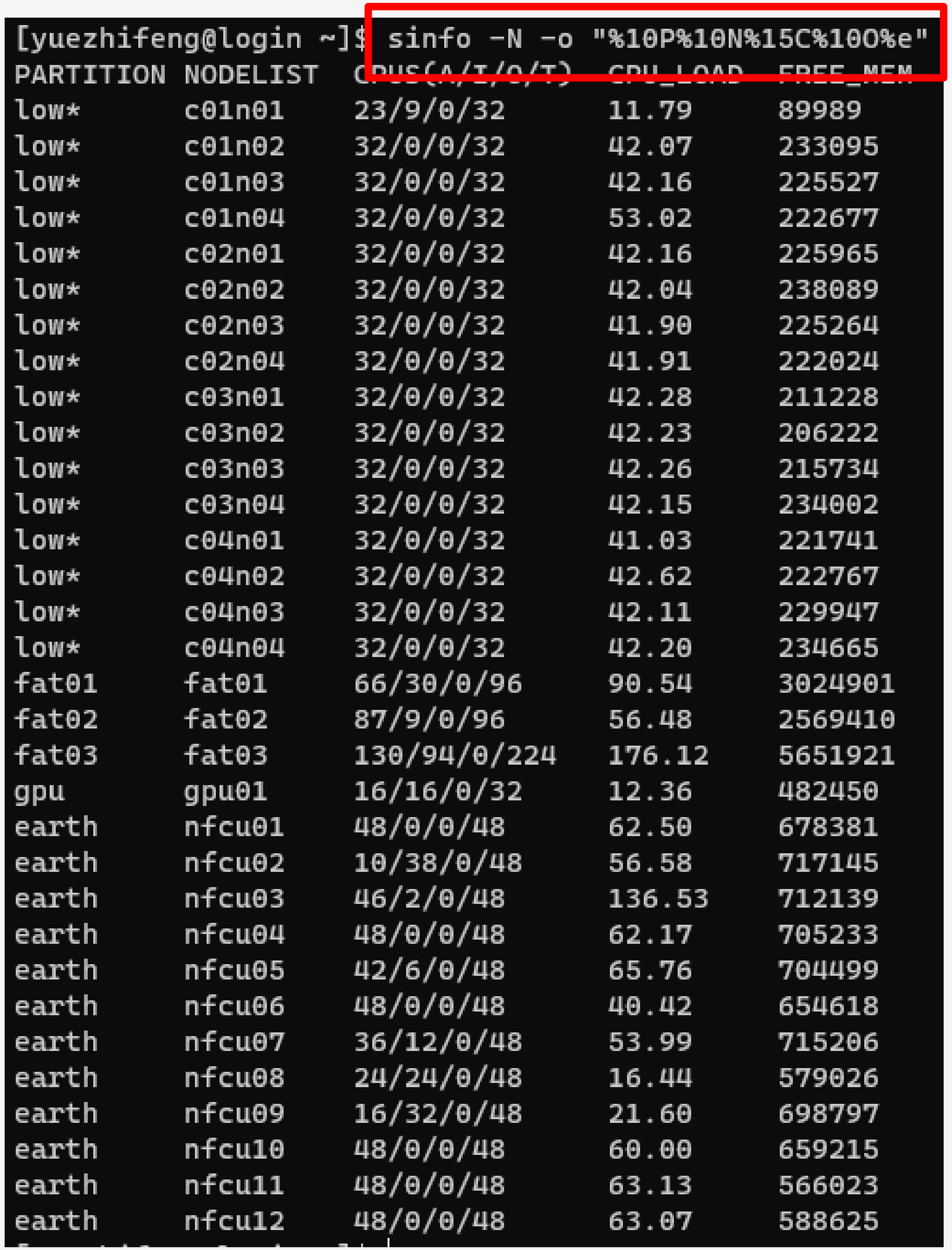
```
[yuezhifeng@login ~]$ sinfo --help
Usage: sinfo [OPTIONS]
  -a, --all                  show all partitions (including hidden and those
                             not accessible)
  -d, --dead                 show only non-responding nodes
  -e, --exact                group nodes only on exact match of configuration
      --federation           Report federated information if a member of one
  -h, --noheader             no headers on output
  --hide                     do not show hidden or non-accessible partitions
  -i, --iterate=seconds      specify an iteration period
      --local                show only local cluster in a federation.
                             Overrides --federation.
  -l, --long                 long output - displays more information
  -M, --clusters=names       clusters to issue commands to. Implies --local.
                             NOTE: SlurmDBD must be up.
  -n, --nodes=NODES          report on specific node(s)
  --noconvert                don't convert units from their original type
                             (e.g. 2048M won't be converted to 2G).
  -N, --Node                 Node-centric format
  -o, --format=format        format specification
  -O, --Format=format        long format specification
  -p, --partition=PARTITION  report on specific partition
  -r, --responding           report only responding nodes
  -R, --list-reasons         list reason nodes are down or drained
  -s, --summarize            report state summary only
  -S, --sort=fields          comma separated list of fields to sort on
  -t, --states=node_state    specify the what states of nodes to view
  -T, --reservation          show only reservation information
  -v, --verbose              verbosity level
  -V, --version              output version information and exit

Help options:
  --help                     show this help message
  --usage                    display brief usage message
```

sinfo

- (A/I/O/T) - allocated/idle/other/total



```
[yuezhifeng@login ~]$ sinfo -O nodehost,cpusstate,cpusload,memory,freemem
HOSTNAMES           CPUS(A/I/O/T)       CPU_LOAD            MEMORY              FREE_MEM
c01n01              23/9/0/32           12.19               230000              85668
c01n02              32/0/0/32           41.89               230000              233034
c01n03              32/0/0/32           42.04               230000              225573
c01n04              32/0/0/32           52.66               230000              222598
c02n01              32/0/0/32           42.09               230000              226028
c02n02              32/0/0/32           41.88               230000              238083
c02n03              32/0/0/32           41.96               230000              225226
c02n04              32/0/0/32           42.10               230000              222031
c03n01              32/0/0/32           42.19               230000              211258
c03n02              32/0/0/32           42.05               230000              206367
c03n03              32/0/0/32           42.12               230000              215663
c03n04              32/0/0/32           42.11               230000              233871
c04n01              32/0/0/32           41.53               230000              221809
c04n02              32/0/0/32           42.25               230000              222799
c04n03              32/0/0/32           41.84               230000              230016
c04n04              32/0/0/32           41.88               230000              234729
fat01               66/30/0/96          89.64               3050000             3024759
fat02               87/9/0/96           56.58               3050000             2569461
fat03               130/94/0/224        171.52              6090000             5649673
gpu01               16/16/0/32          12.91               510000              481784
nfcu02              10/38/0/48          59.66               745000              716956
nfcu03              46/2/0/48           132.04              745000              712347
nfcu05              42/6/0/48           63.40               745000              704570
nfcu07              36/12/0/48          50.97               745000              715276
nfcu08              24/24/0/48          16.30               745000              579082
nfcu09              16/32/0/48          19.64               745000              697858
nfcu01              48/0/0/48           60.60               745000              678385
nfcu04              48/0/0/48           62.96               745000              705233
nfcu06              48/0/0/48           57.77               745000              654652
nfcu10              48/0/0/48           58.98               745000              659212
nfcu11              48/0/0/48           63.66               745000              566017
nfcu12              48/0/0/48           63.44               745000              588599
```

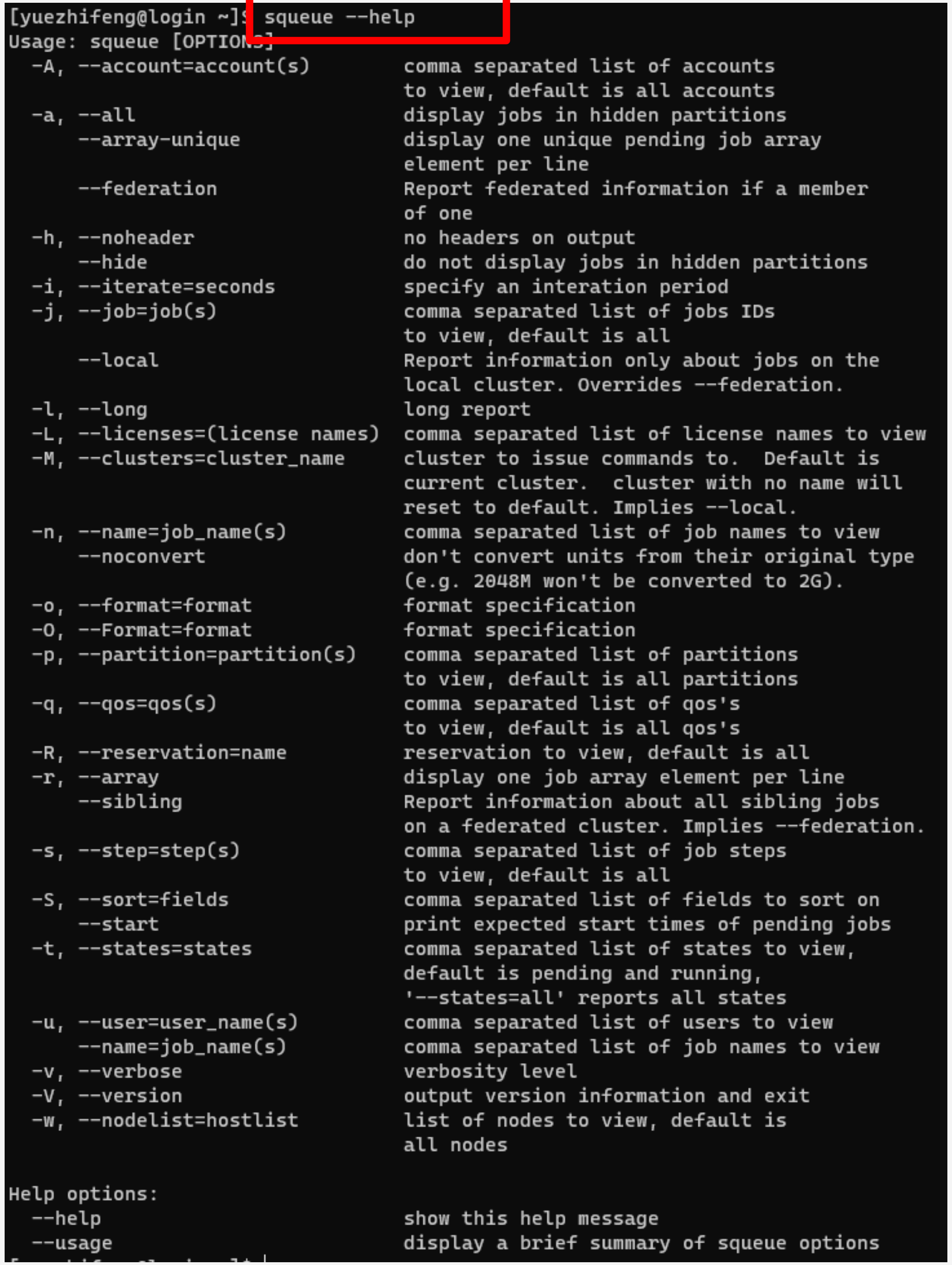


```
[yuezhifeng@login ~]$ sinfo -N -o "%10P%10N%15C%10O%e"
PARTITION NODELIST  CPUS(A/I/O/T)  CPU_LOAD  FREE_MEM
low*      c01n01    23/9/0/32      11.79     89989
low*      c01n02    32/0/0/32      42.07     233095
low*      c01n03    32/0/0/32      42.16     225527
low*      c01n04    32/0/0/32      53.02     222677
low*      c02n01    32/0/0/32      42.16     225965
low*      c02n02    32/0/0/32      42.04     238089
low*      c02n03    32/0/0/32      41.90     225264
low*      c02n04    32/0/0/32      41.91     222024
low*      c03n01    32/0/0/32      42.28     211228
low*      c03n02    32/0/0/32      42.23     206222
low*      c03n03    32/0/0/32      42.26     215734
low*      c03n04    32/0/0/32      42.15     234002
low*      c04n01    32/0/0/32      41.03     221741
low*      c04n02    32/0/0/32      42.62     222767
low*      c04n03    32/0/0/32      42.11     229947
low*      c04n04    32/0/0/32      42.20     234665
fat01     fat01     66/30/0/96     90.54     3024901
fat02     fat02     87/9/0/96      56.48     2569410
fat03     fat03     130/94/0/224   176.12    5651921
gpu       gpu01     16/16/0/32     12.36     482450
earth     nfcu01    48/0/0/48      62.50     678381
earth     nfcu02    10/38/0/48     56.58     717145
earth     nfcu03    46/2/0/48      136.53    712139
earth     nfcu04    48/0/0/48      62.17     705233
earth     nfcu05    42/6/0/48      65.76     704499
earth     nfcu06    48/0/0/48      40.42     654618
earth     nfcu07    36/12/0/48     53.99     715206
earth     nfcu08    24/24/0/48     16.44     579026
earth     nfcu09    16/32/0/48     21.60     698797
earth     nfcu10    48/0/0/48      60.00     659215
earth     nfcu11    48/0/0/48      63.13     566023
earth     nfcu12    48/0/0/48      63.07     588625
```

squeue

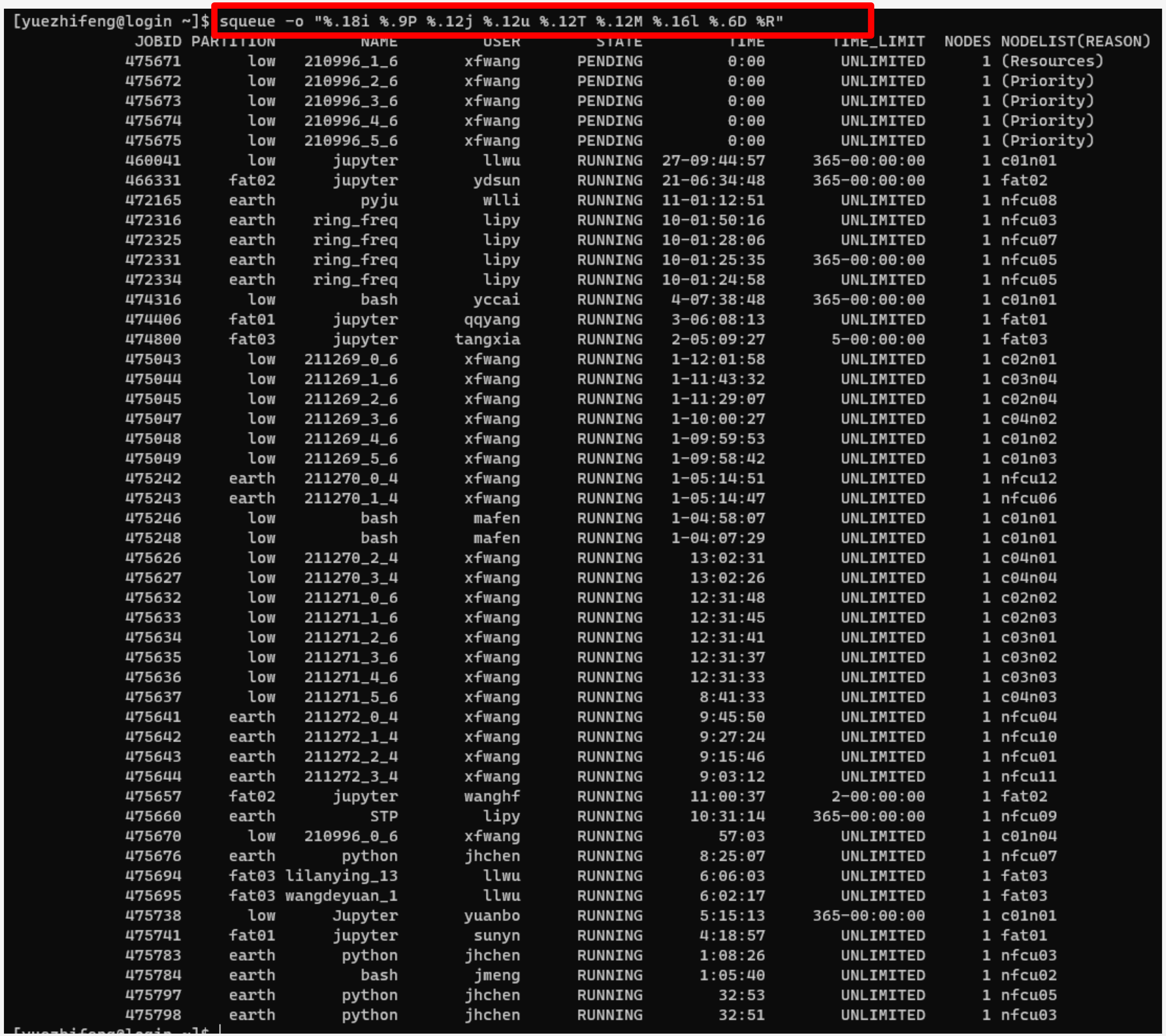
- 学习squeue最好的方法就是看help

- https://slurm.schedmd.com/squeue.html
- http://hpc.pku.edu.cn/_book/guide/slurm/squeue.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#squeue



```
[yuezhifeng@login ~]$ squeue --help
Usage: squeue [OPTIONS]
  -A, --account=account(s)        comma separated list of accounts
                                  to view, default is all accounts
  -a, --all                       display jobs in hidden partitions
      --array-unique              display one unique pending job array
                                  element per line
      --federation                Report federated information if a member
                                  of one
  -h, --noheader                  no headers on output
      --hide                      do not display jobs in hidden partitions
  -i, --iterate=seconds           specify an interation period
  -j, --job=job(s)                comma separated list of jobs IDs
                                  to view, default is all
      --local                     Report information only about jobs on the
                                  local cluster. Overrides --federation.
  -l, --long                      long report
  -L, --licenses=(license names)  comma separated list of license names to view
  -M, --clusters=cluster_name     cluster to issue commands to.  Default is
                                  current cluster.  cluster with no name will
                                  reset to default. Implies --local.
  -n, --name=job_name(s)          comma separated list of job names to view
      --noconvert                 don't convert units from their original type
                                  (e.g. 2048M won't be converted to 2G).
  -o, --format=format             format specification
  -O, --Format=format             format specification
  -p, --partition=partition(s)    comma separated list of partitions
                                  to view, default is all partitions
  -q, --qos=qos(s)                comma separated list of qos's
                                  to view, default is all qos's
  -R, --reservation=name          reservation to view, default is all
  -r, --array                     display one job array element per line
      --sibling                   Report information about all sibling jobs
                                  on a federated cluster. Implies --federation.
  -s, --step=step(s)              comma separated list of job steps
                                  to view, default is all
  -S, --sort=fields               comma separated list of fields to sort on
      --start                     print expected start times of pending jobs
  -t, --states=states             comma separated list of states to view,
                                  default is pending and running,
                                  '--states=all' reports all states
  -u, --user=user_name(s)         comma separated list of users to view
      --name=job_name(s)          comma separated list of job names to view
  -v, --verbose                   verbosity level
  -V, --version                   output version information and exit
  -w, --nodelist=hostlist         list of nodes to view, default is
                                  all nodes


Help options:
  --help                          show this help message
  --usage                         display a brief summary of squeue options
```

squeue



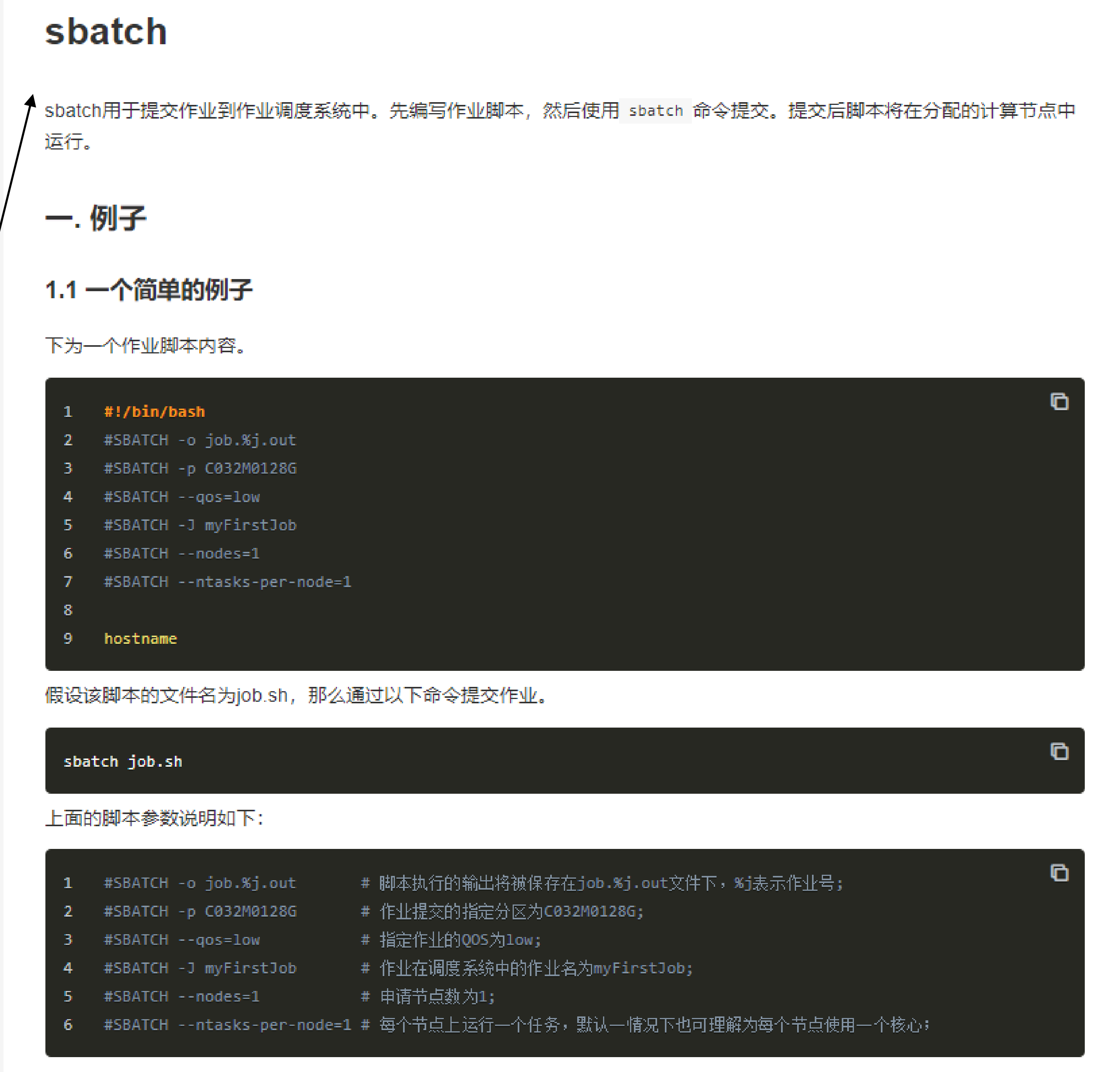
```
[yuezhifeng@login ~]$ squeue -o "%.18i %.9P %.12j %.12u %.12T %.12M %.16l %.6D %R"
             JOBID PARTITION         NAME         USER        STATE         TIME       TIME_LIMIT  NODES NODELIST(REASON)
            475671       low   210996_1_6       xfwang      PENDING         0:00        UNLIMITED      1 (Resources)
            475672       low   210996_2_6       xfwang      PENDING         0:00        UNLIMITED      1 (Priority)
            475673       low   210996_3_6       xfwang      PENDING         0:00        UNLIMITED      1 (Priority)
            475674       low   210996_4_6       xfwang      PENDING         0:00        UNLIMITED      1 (Priority)
            475675       low   210996_5_6       xfwang      PENDING         0:00        UNLIMITED      1 (Priority)
            460041       low      jupyter         llwu      RUNNING  27-09:44:57     365-00:00:00      1 c01n01
            466331     fat02      jupyter        ydsun      RUNNING  21-06:34:48     365-00:00:00      1 fat02
            472165     earth         pyju         wlli      RUNNING  11-01:12:51        UNLIMITED      1 nfcu08
            472316     earth    ring_freq         lipy      RUNNING  10-01:50:16        UNLIMITED      1 nfcu03
            472325     earth    ring_freq         lipy      RUNNING  10-01:28:06        UNLIMITED      1 nfcu07
            472331     earth    ring_freq         lipy      RUNNING  10-01:25:35     365-00:00:00      1 nfcu05
            472334     earth    ring_freq         lipy      RUNNING  10-01:24:58        UNLIMITED      1 nfcu05
            474316       low         bash        yccai      RUNNING   4-07:38:48     365-00:00:00      1 c01n01
            474406     fat01      jupyter       qqyang      RUNNING   3-06:08:13        UNLIMITED      1 fat01
            474800     fat03      jupyter      tangxia      RUNNING   2-05:09:27       5-00:00:00      1 fat03
            475043       low   211269_0_6       xfwang      RUNNING   1-12:01:58        UNLIMITED      1 c02n01
            475044       low   211269_1_6       xfwang      RUNNING   1-11:43:32        UNLIMITED      1 c03n04
            475045       low   211269_2_6       xfwang      RUNNING   1-11:29:07        UNLIMITED      1 c02n04
            475047       low   211269_3_6       xfwang      RUNNING   1-10:00:27        UNLIMITED      1 c04n02
            475048       low   211269_4_6       xfwang      RUNNING   1-09:59:53        UNLIMITED      1 c01n02
            475049       low   211269_5_6       xfwang      RUNNING   1-09:58:42        UNLIMITED      1 c01n03
            475242     earth   211270_0_4       xfwang      RUNNING   1-05:14:51        UNLIMITED      1 nfcu12
            475243     earth   211270_1_4       xfwang      RUNNING   1-05:14:47        UNLIMITED      1 nfcu06
            475246       low         bash        mafen      RUNNING   1-04:58:07        UNLIMITED      1 c01n01
            475248       low         bash        mafen      RUNNING   1-04:07:29        UNLIMITED      1 c01n01
            475626       low   211270_2_4       xfwang      RUNNING     13:02:31        UNLIMITED      1 c04n01
            475627       low   211270_3_4       xfwang      RUNNING     13:02:26        UNLIMITED      1 c04n04
            475632       low   211271_0_6       xfwang      RUNNING     12:31:48        UNLIMITED      1 c02n02
            475633       low   211271_1_6       xfwang      RUNNING     12:31:45        UNLIMITED      1 c02n03
            475634       low   211271_2_6       xfwang      RUNNING     12:31:41        UNLIMITED      1 c03n01
            475635       low   211271_3_6       xfwang      RUNNING     12:31:37        UNLIMITED      1 c03n02
            475636       low   211271_4_6       xfwang      RUNNING     12:31:33        UNLIMITED      1 c03n03
            475637       low   211271_5_6       xfwang      RUNNING      8:41:33        UNLIMITED      1 c04n03
            475641     earth   211272_0_4       xfwang      RUNNING      9:45:50        UNLIMITED      1 nfcu04
            475642     earth   211272_1_4       xfwang      RUNNING      9:27:24        UNLIMITED      1 nfcu10
            475643     earth   211272_2_4       xfwang      RUNNING      9:15:46        UNLIMITED      1 nfcu01
            475644     earth   211272_3_4       xfwang      RUNNING      9:03:12        UNLIMITED      1 nfcu11
            475657     fat02      jupyter       wanghf      RUNNING     11:00:37       2-00:00:00      1 fat02
            475660     earth          STP         lipy      RUNNING     10:31:14     365-00:00:00      1 nfcu09
            475670       low   210996_0_6       xfwang      RUNNING        57:03        UNLIMITED      1 c01n04
            475676     earth       python       jhchen      RUNNING      8:25:07        UNLIMITED      1 nfcu07
            475694     fat03 lilanying_13         llwu      RUNNING      6:06:03        UNLIMITED      1 fat03
            475695     fat03 wangdeyuan_1         llwu      RUNNING      6:02:17        UNLIMITED      1 fat03
            475738       low      Jupyter       yuanbo      RUNNING      5:15:13     365-00:00:00      1 c01n01
            475741     fat01      jupyter        sunyn      RUNNING      4:18:57        UNLIMITED      1 fat01
            475783     earth       python       jhchen      RUNNING      1:08:26        UNLIMITED      1 nfcu03
            475784     earth         bash        jmeng      RUNNING      1:05:40        UNLIMITED      1 nfcu02
            475797     earth       python       jhchen      RUNNING        32:53        UNLIMITED      1 nfcu05
            475798     earth       python       jhchen      RUNNING        32:51        UNLIMITED      1 nfcu03
```

©2020-2021 中国科学院脑智卓越中心-脑科学数据与计算中心 版权所有

sbatch

- 学习sbatch最好的方法就是看help

- https://slurm.schedmd.com/sbatch.html
- http://hpc.pku.edu.cn/_book/guide/slurm/sbatch.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#sbatch



sbatch

sbatch用于提交作业到作业调度系统中。先编写作业脚本，然后使用 `sbatch` 命令提交。提交后脚本将在分配的计算节点中运行。

一. 例子

1.1 一个简单的例子

下为一个作业脚本内容。

```
#!/bin/bash
#SBATCH -o job.%j.out
#SBATCH -p C032M0128G
#SBATCH --qos=low
#SBATCH -J myFirstJob
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1

hostname
```

假设该脚本的文件名为job.sh，那么通过以下命令提交作业。

```
sbatch job.sh
```

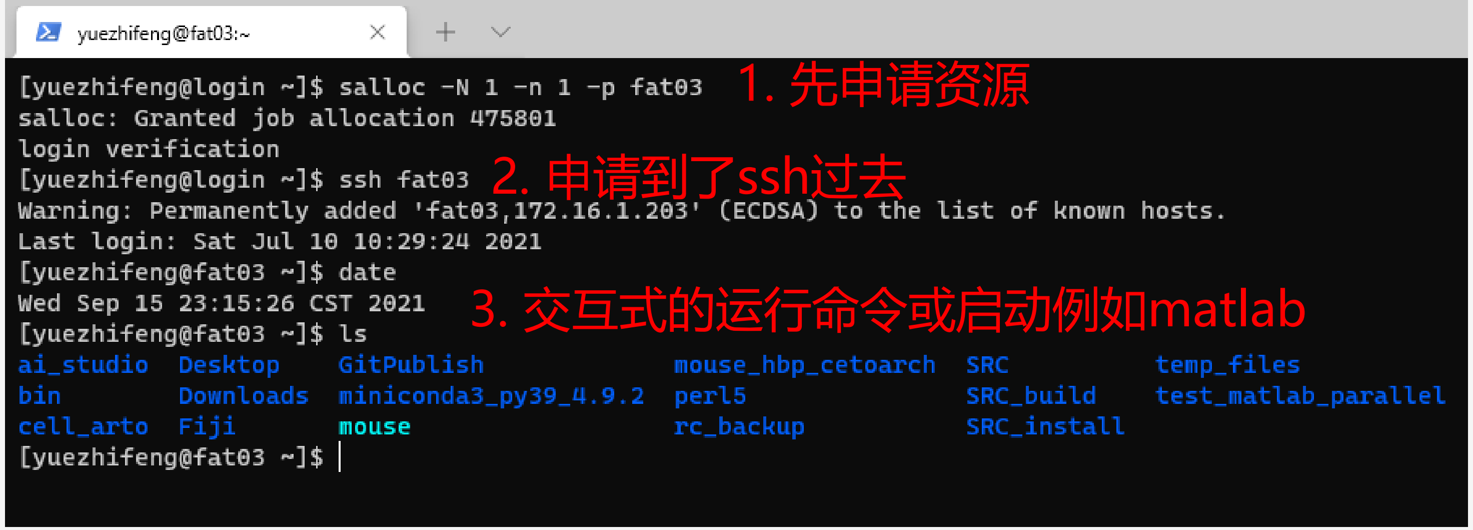
上面的脚本参数说明如下：

```
#SBATCH -o job.%j.out       # 脚本执行的输出将被保存在job.%j.out文件下，%j表示作业号;
#SBATCH -p C032M0128G       # 作业提交的指定分区为C032M0128G;
#SBATCH --qos=low           # 指定作业的QOS为low;
#SBATCH -J myFirstJob       # 作业在调度系统中的作业名为myFirstJob;
#SBATCH --nodes=1           # 申请节点数为1;
#SBATCH --ntasks-per-node=1 # 每个节点上运行一个任务，默认一情况下也可理解为每个节点使用一个核心;
```

salloc

- 学习salloc最好的方法就是看help

- https://slurm.schedmd.com/salloc.html
- http://hpc.pku.edu.cn/_book/guide/slurm/salloc.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#srun-salloc



```
[yuezhifeng@login ~]$ salloc -N 1 -n 1 -p fat03
salloc: Granted job allocation 475801
login verification
[yuezhifeng@login ~]$ ssh fat03
Warning: Permanently added 'fat03,172.16.1.203' (ECDSA) to the list of known hosts.
Last login: Sat Jul 10 10:29:24 2021
[yuezhifeng@fat03 ~]$ date
Wed Sep 15 23:15:26 CST 2021
[yuezhifeng@fat03 ~]$ ls
ai_studio  Desktop    GitPublish             mouse_hbp_cetoarch  SRC          temp_files
bin        Downloads  miniconda3_py39_4.9.2  perl5               SRC_build    test_matlab_parallel
cell_arto  Fiji       mouse                  rc_backup           SRC_install
[yuezhifeng@fat03 ~]$
```

1. 先申请资源
2. 申请到了ssh过去
3. 交互式的运行命令或启动例如matlab

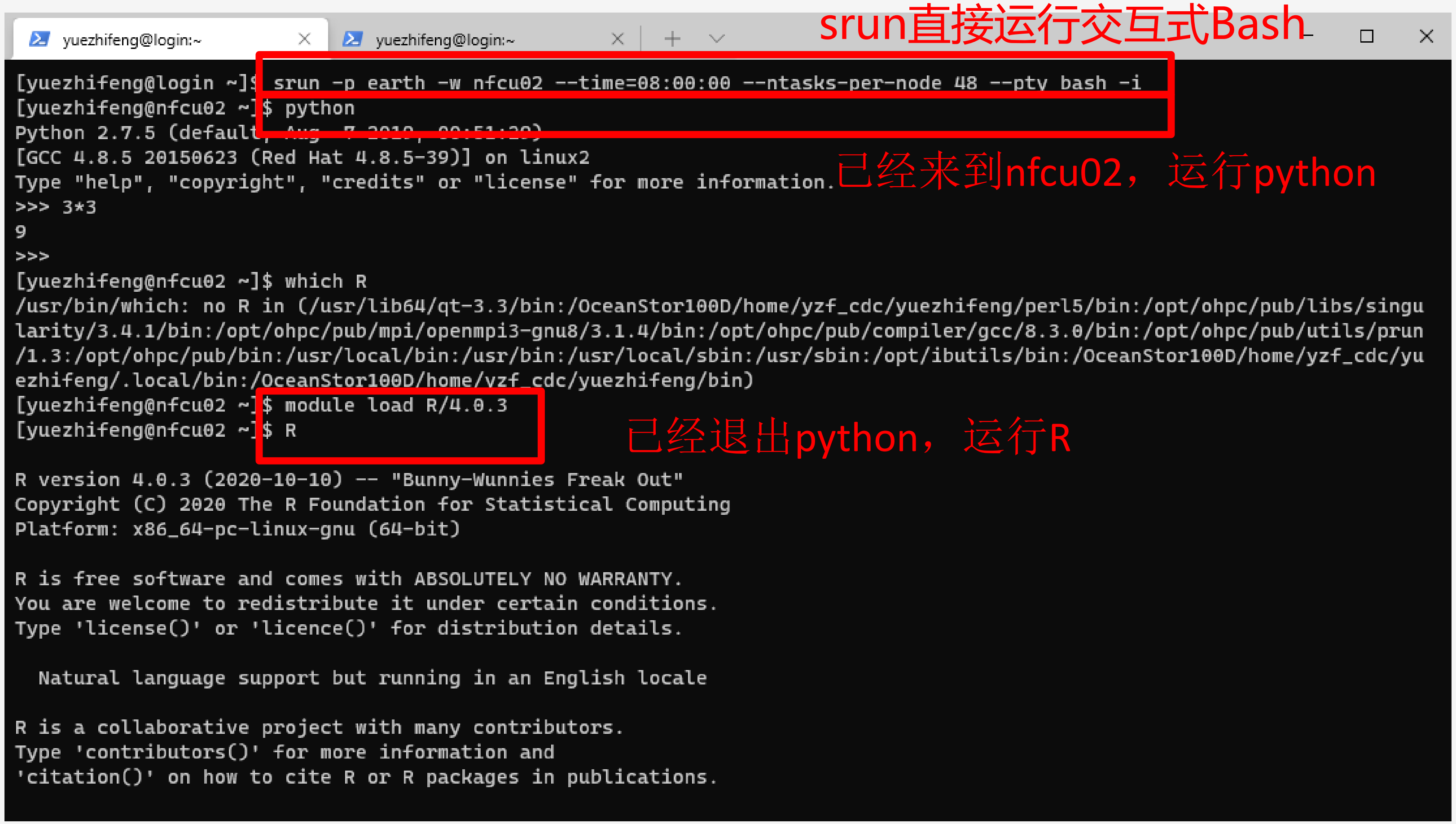
srun

- 学习srun最好的方法就是看help

- https://slurm.schedmd.com/srun.html

- http://hpc.pku.edu.cn/_book/guide/slurm/srun.html

- https://docs.hpc.sjtu.edu.cn/job/slurm.html#srun-salloc



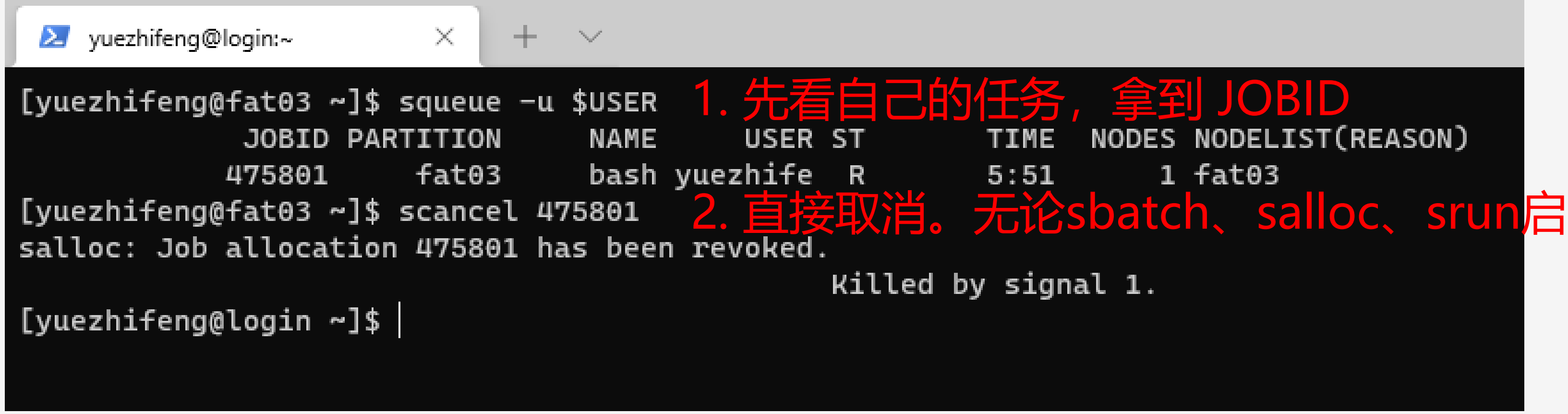
srun直接运行交互式Bash

已经来到nfcu02，运行python

已经退出python，运行R

```
[yuezhifeng@login ~]$ srun -p earth -w nfcu02 --time=08:00:00 --ntasks-per-node 48 --pty bash -i
[yuezhifeng@nfcu02 ~]$ python
Python 2.7.5 (default, Aug  7 2019, 00:51:29)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-39)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 3*3
9
>>>
[yuezhifeng@nfcu02 ~]$ which R
/usr/bin/which: no R in (/usr/lib64/qt-3.3/bin:/OceanStor100D/home/yzf_cdc/yuezhifeng/perl5/bin:/opt/ohpc/pub/libs/singu
larity/3.4.1/bin:/opt/ohpc/pub/mpi/openmpi3-gnu8/3.1.4/bin:/opt/ohpc/pub/compiler/gcc/8.3.0/bin:/opt/ohpc/pub/utils/prun
/1.3:/opt/ohpc/pub/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/opt/ibutils/bin:/OceanStor100D/home/yzf_cdc/yu
ezhifeng/.local/bin:/OceanStor100D/home/yzf_cdc/yuezhifeng/bin)
[yuezhifeng@nfcu02 ~]$ module load R/4.0.3
[yuezhifeng@nfcu02 ~]$ R

R version 4.0.3 (2020-10-10) -- "Bunny-Wunnies Freak Out"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
```

scancel



```
[yuezhifeng@fat03 ~]$ squeue -u $USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
            475801     fat03     bash yuezhife  R       5:51      1 fat03
[yuezhifeng@fat03 ~]$ scancel 475801
salloc: Job allocation 475801 has been revoked.
                                               Killed by signal 1.

[yuezhifeng@login ~]$ 
```

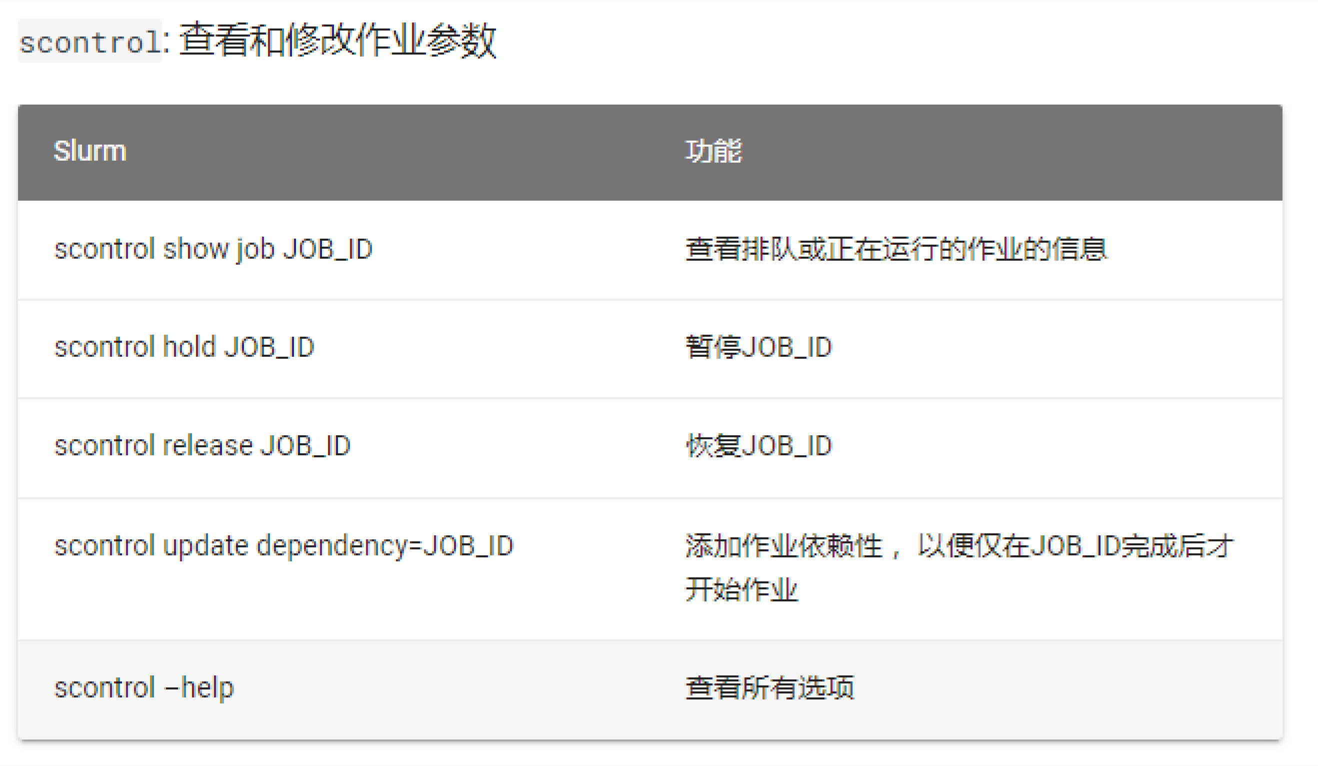
1. 先看自己的任务，拿到 JOBID

2. 直接取消。无论sbatch、salloc、srun启动的任务的都能这么取消

scontrol

- 学习scontrol最好的方法就是看help

- https://slurm.schedmd.com/scontrol.html
- http://hpc.pku.edu.cn/_book/guide/slurm/sacct.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#srun-salloc



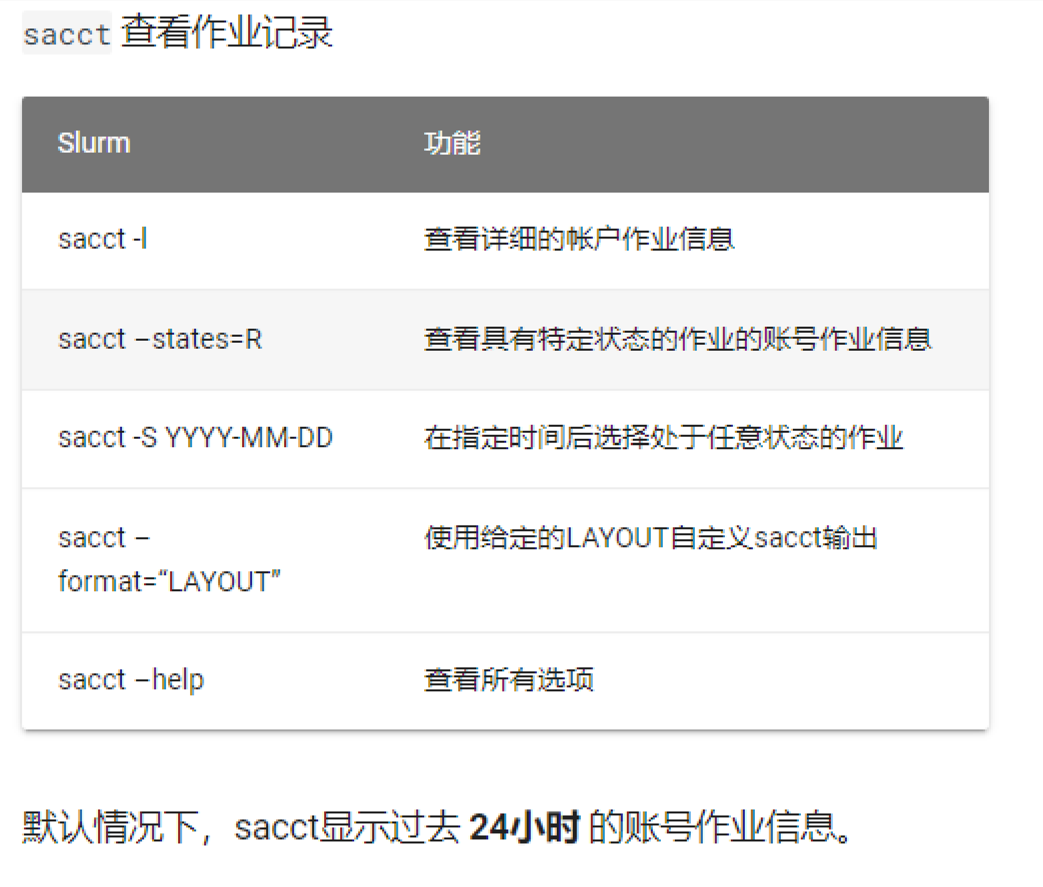
`scontrol`: 查看和修改作业参数

Slurm	功能
scontrol show job JOB_ID	查看排队或正在运行的作业的信息
scontrol hold JOB_ID	暂停JOB_ID
scontrol release JOB_ID	恢复JOB_ID
scontrol update dependency=JOB_ID	添加作业依赖性， 以便仅在JOB_ID完成后才开始作业
scontrol –help	查看所有选项

sacct

- 学习scontrol最好的方法就是看help

- https://slurm.schedmd.com/sacct.html
- http://hpc.pku.edu.cn/_book/guide/slurm/sacct.html
- https://docs.hpc.sjtu.edu.cn/job/slurm.html#srun-salloc



`sacct` 查看作业记录

Slurm	功能
sacct -l	查看详细的帐户作业信息
sacct –states=R	查看具有特定状态的作业的账号作业信息
sacct -S YYYY-MM-DD	在指定时间后选择处于任意状态的作业
sacct –format=“LAYOUT”	使用给定的LAYOUT自定义sacct输出
sacct –help	查看所有选项

默认情况下，sacct显示过去 **24小时** 的账号作业信息。

一 使用集群计算资源的方式

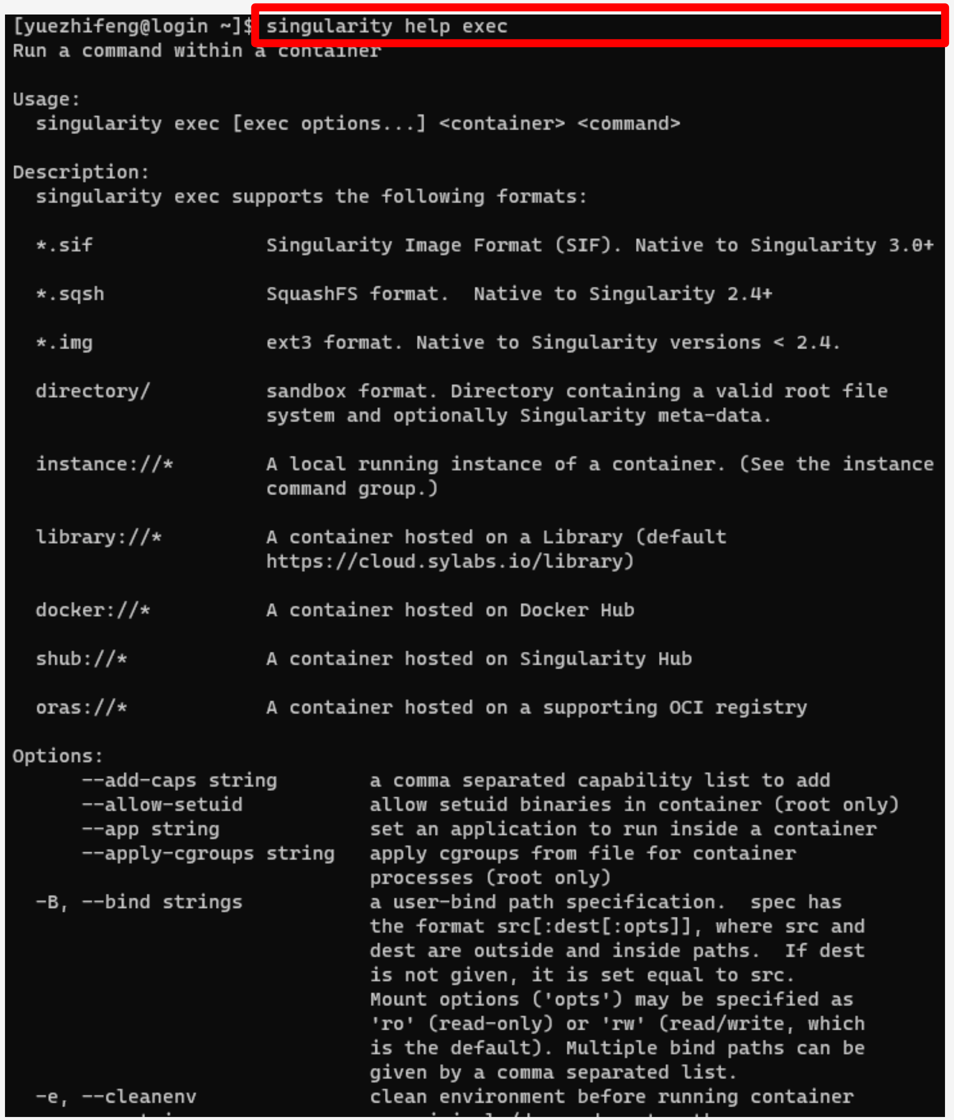
二 集群硬件和软件栈介绍

三 CPU集群容器工具介绍与容器示例

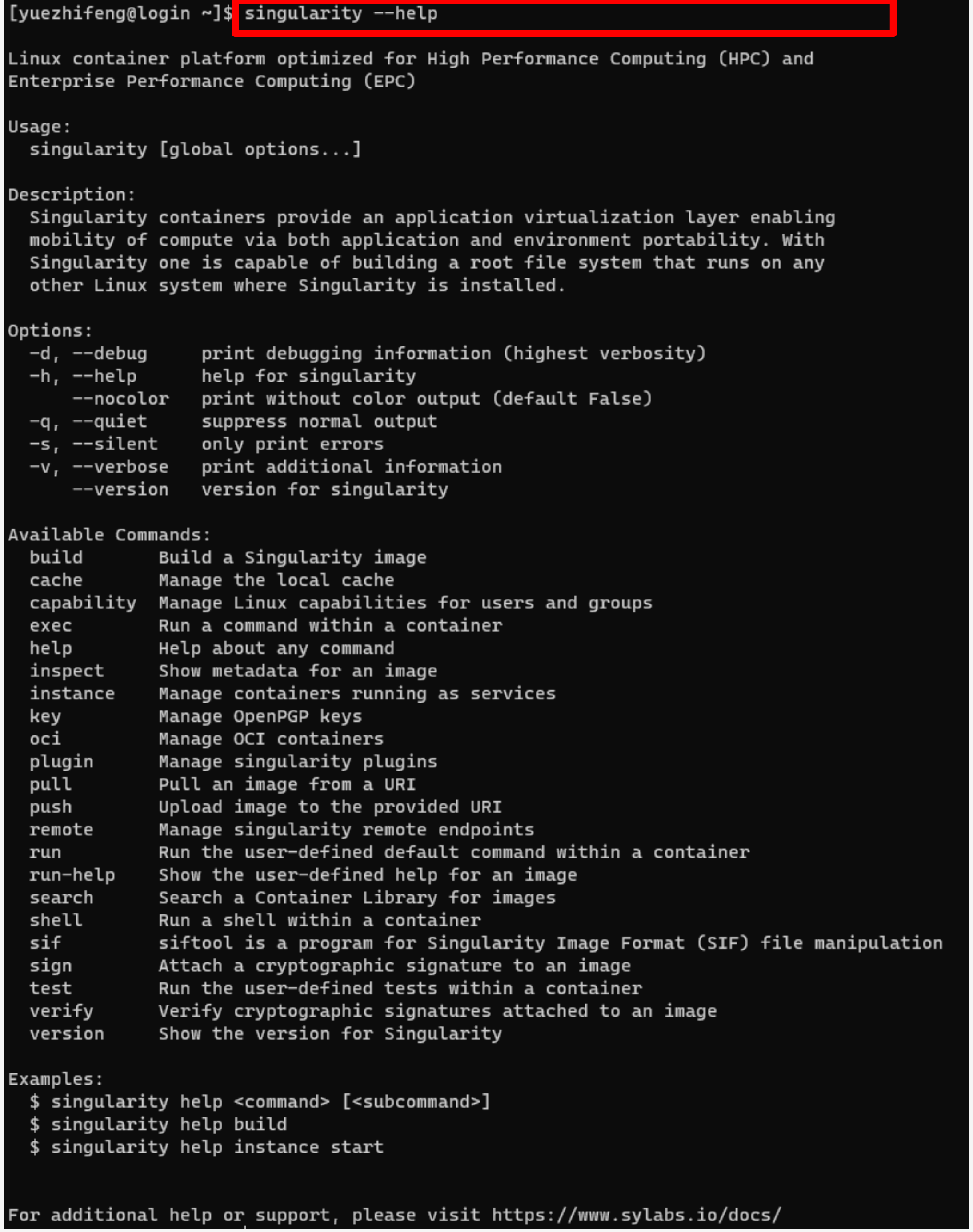
四 交互式使用计算资源的详细步骤

五 集群使用注意事项和账户申请流程

学习singularity



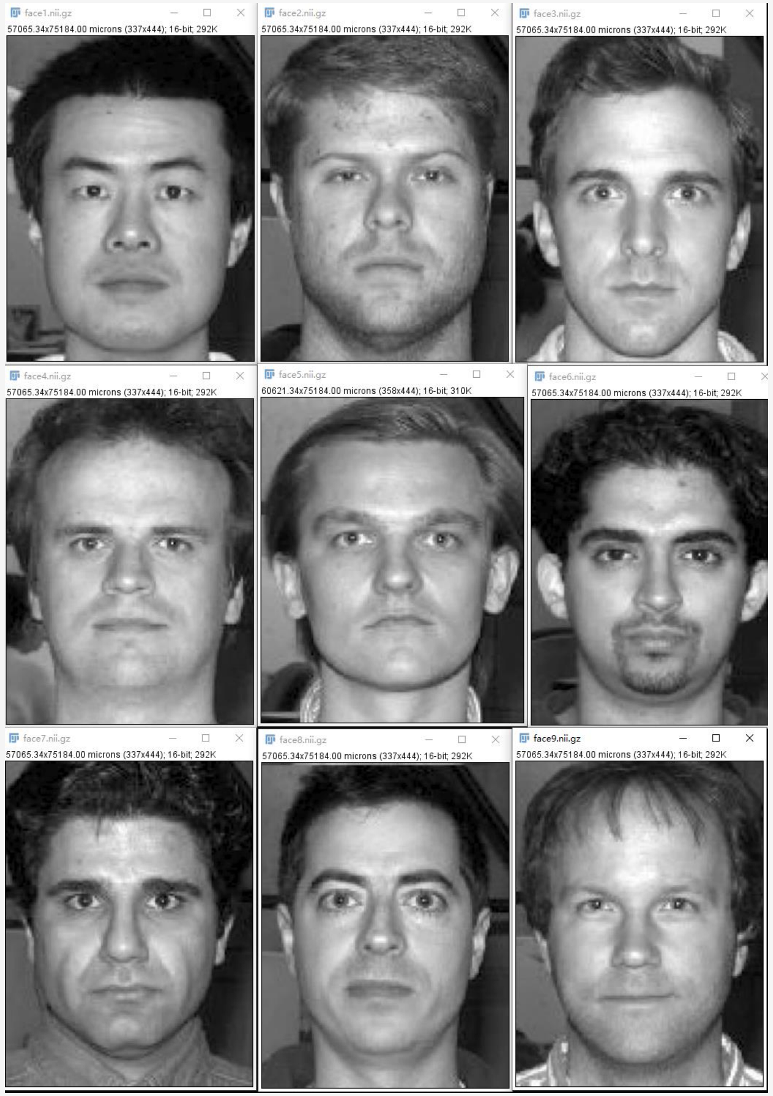
```
[yuezhifeng@login ~]$ singularity help exec
Run a command within a container

Usage:
  singularity exec [exec options...] <container> <command>

Description:
  singularity exec supports the following formats:

  *.sif               Singularity Image Format (SIF). Native to Singularity 3.0+

  *.sqsh              SquashFS format.  Native to Singularity 2.4+

  *.img               ext3 format. Native to Singularity versions < 2.4.

  directory/          sandbox format. Directory containing a valid root file
                      system and optionally Singularity meta-data.

  instance://*        A local running instance of a container. (See the instance
                      command group.)

  library://*         A container hosted on a Library (default
                      https://cloud.sylabs.io/library)

  docker://*          A container hosted on Docker Hub

  shub://*            A container hosted on Singularity Hub

  oras://*            A container hosted on a supporting OCI registry

Options:
      --add-caps string        a comma separated capability list to add
      --allow-setuid           allow setuid binaries in container (root only)
      --app string             set an application to run inside a container
      --apply-cgroups string   apply cgroups from file for container
                               processes (root only)
  -B, --bind strings           a user-bind path specification.  spec has
                               the format src[:dest[:opts]], where src and
                               dest are outside and inside paths.  If dest
                               is not given, it is set equal to src.
                               Mount options ('opts') may be specified as
                               'ro' (read-only) or 'rw' (read/write, which
                               is the default). Multiple bind paths can be
                               given by a comma separated list.
  -e, --cleanenv               clean environment before running container
```



```
[yuezhifeng@login ~]$ singularity --help

Linux container platform optimized for High Performance Computing (HPC) and
Enterprise Performance Computing (EPC)

Usage:
  singularity [global options...]

Description:
  Singularity containers provide an application virtualization layer enabling
  mobility of compute via both application and environment portability. With
  Singularity one is capable of building a root file system that runs on any
  other Linux system where Singularity is installed.

Options:
  -d, --debug     print debugging information (highest verbosity)
  -h, --help      help for singularity
      --nocolor   print without color output (default False)
  -q, --quiet     suppress normal output
  -s, --silent    only print errors
  -v, --verbose   print additional information
      --version   version for singularity

Available Commands:
  build       Build a Singularity image
  cache       Manage the local cache
  capability  Manage Linux capabilities for users and groups
  exec        Run a command within a container
  help        Help about any command
  inspect     Show metadata for an image
  instance    Manage containers running as services
  key         Manage OpenPGP keys
  oci         Manage OCI containers
  plugin      Manage singularity plugins
  pull        Pull an image from a URI
  push        Upload image to the provided URI
  remote      Manage singularity remote endpoints
  run         Run the user-defined default command within a container
  run-help    Show the user-defined help for an image
  search      Search a Container Library for images
  shell       Run a shell within a container
  sif         siftool is a program for Singularity Image Format (SIF) file manipulation
  sign        Attach a cryptographic signature to an image
  test        Run the user-defined tests within a container
  verify      Verify cryptographic signatures attached to an image
  version     Show the version for Singularity

Examples:
  $ singularity help <command> [<subcommand>]
  $ singularity help build
  $ singularity help instance start


For additional help or support, please visit https://www.sylabs.io/docs/
```

学习使用ants的singularity镜像制作平均脸

目标：用右边9张脸制作平均脸



face1.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face2.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face3.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face4.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face5.nii.gz
60621.34x75184.00 microns (358x444); 16-bit; 310K

face6.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

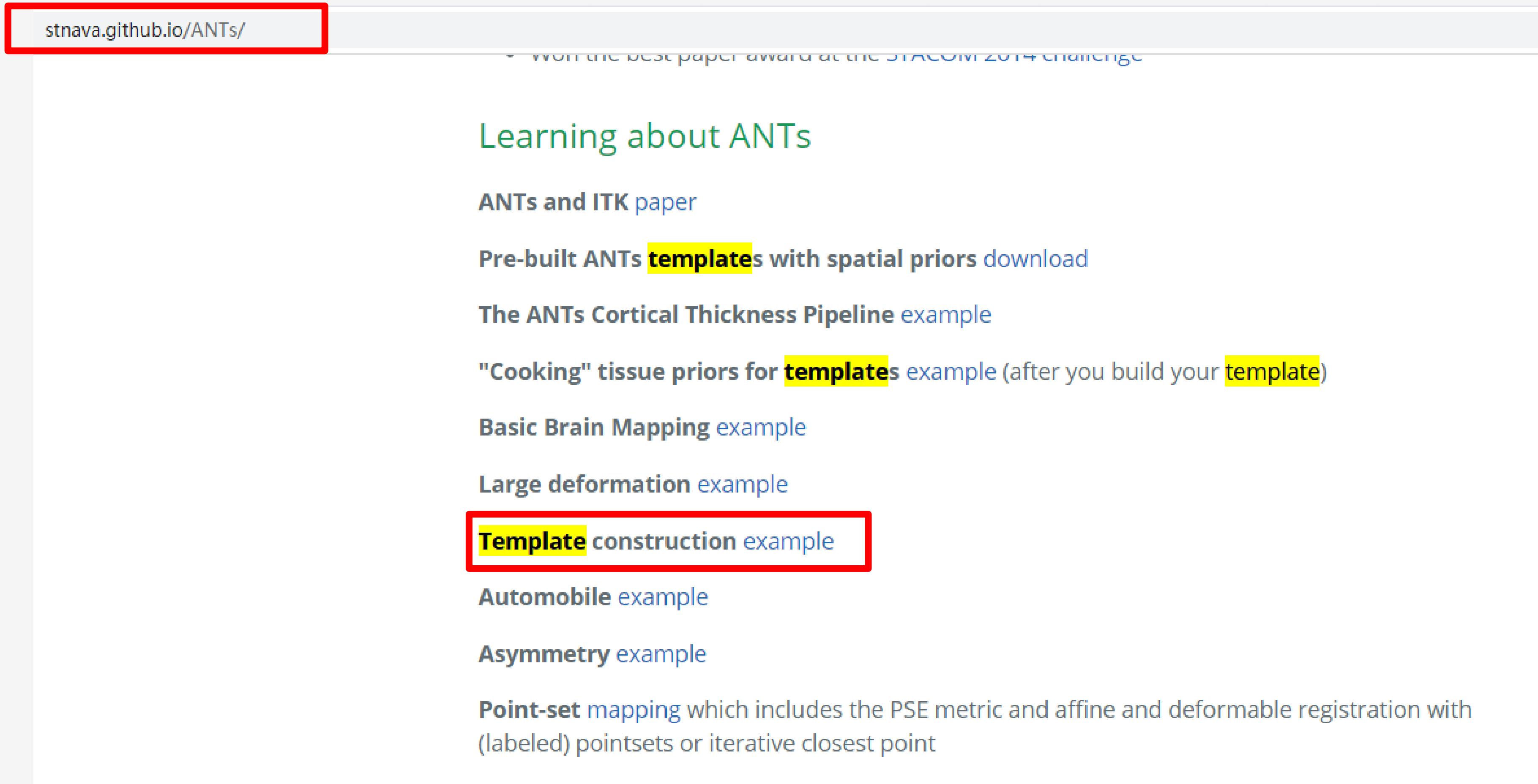
face7.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face8.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

face9.nii.gz
57065.34x75184.00 microns (337x444); 16-bit; 292K

学习使用ants的singularity镜像制作平均脸

Step 1. 下载数据到个人电脑



stnava.github.io/ANTs/

- Won the best paper award at the STACOM 2014 challenge

Learning about ANTs

ANTs and ITK paper

Pre-built ANTs templates with spatial priors download

The ANTs Cortical Thickness Pipeline example

"Cooking" tissue priors for templates example (after you build your template)

Basic Brain Mapping example

Large deformation example

Template construction example

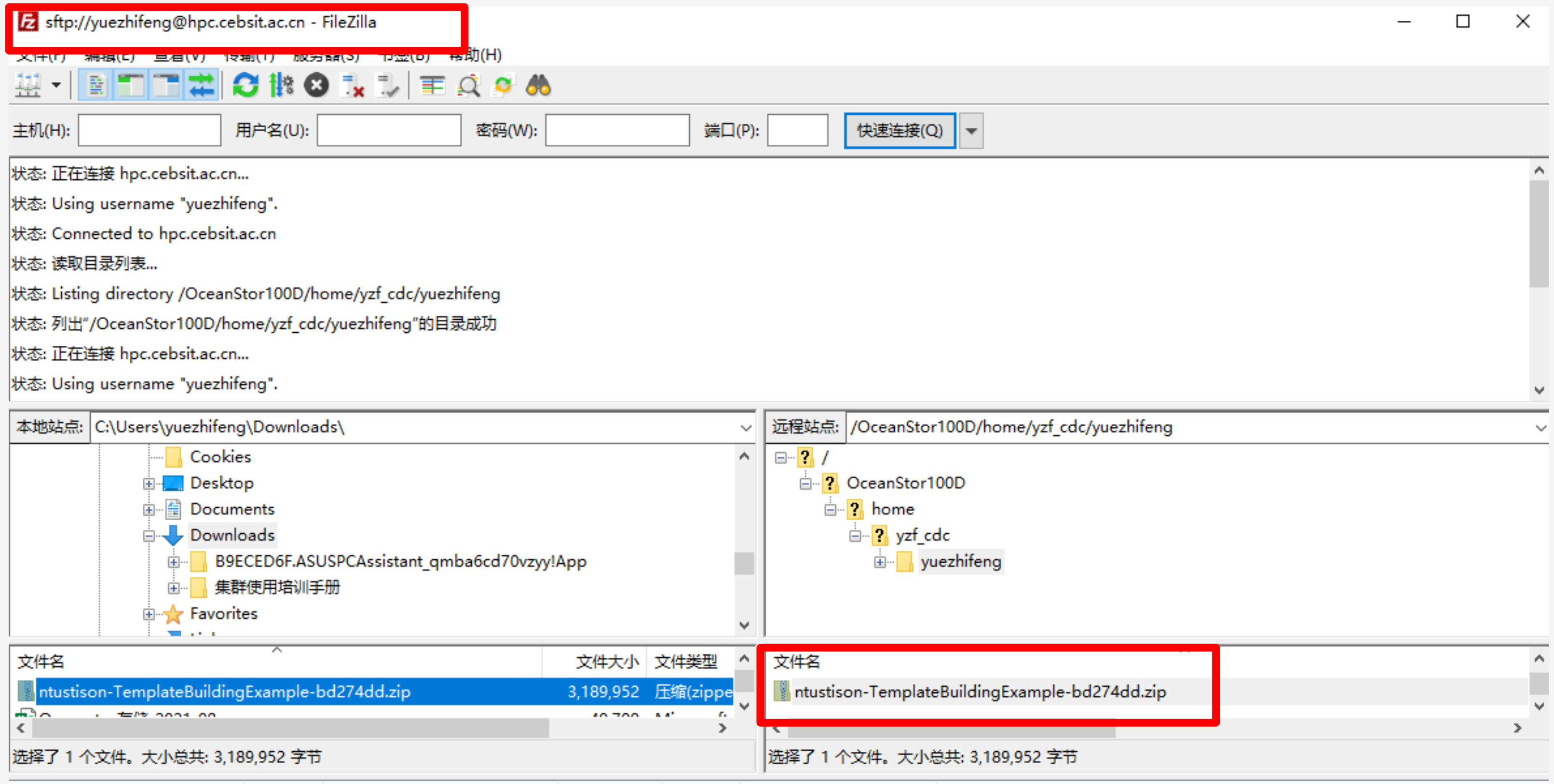
Automobile example

Asymmetry example

Point-set mapping which includes the PSE metric and affine and deformable registration with (labeled) pointsets or iterative closest point

学习使用ants的singularity镜像制作平均脸

Step 2. 从个人电脑上传数据到集群



sftp://yuezhifeng@hpc.cebsit.ac.cn - FileZilla

```
状态: 正在连接 hpc.cebsit.ac.cn...
状态: Using username "yuezhifeng".
状态: Connected to hpc.cebsit.ac.cn
状态: 读取目录列表...
状态: Listing directory /OceanStor100D/home/yzf_cdc/yuezhifeng
状态: 列出"/OceanStor100D/home/yzf_cdc/yuezhifeng"的目录成功
状态: 正在连接 hpc.cebsit.ac.cn...
状态: Using username "yuezhifeng".
```

本地站点: C:\Users\yuezhifeng\Downloads\

- Cookies
- Desktop
- Documents
- Downloads
  - B9ECED6F.ASUSPCAssistant_qmba6cd70vzyy!App
  - 集群使用培训手册
- Favorites

文件名	文件大小	文件类型
ntustison-TemplateBuildingExample-bd274dd.zip	3,189,952	压缩(zippe

选择了 1 个文件。大小总共: 3,189,952 字节

远程站点: /OceanStor100D/home/yzf_cdc/yuezhifeng

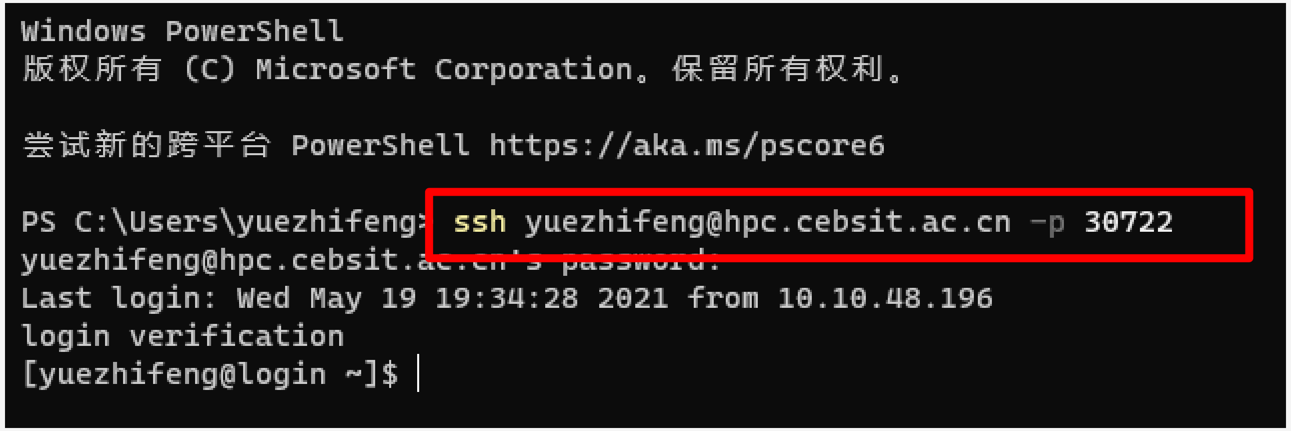
- /
  - OceanStor100D
    - home
      - yzf_cdc
        - yuezhifeng

文件名
ntustison-TemplateBuildingExample-bd274dd.zip

选择了 1 个文件。大小总共: 3,189,952 字节

学习使用ants的singularity镜像制作平均脸

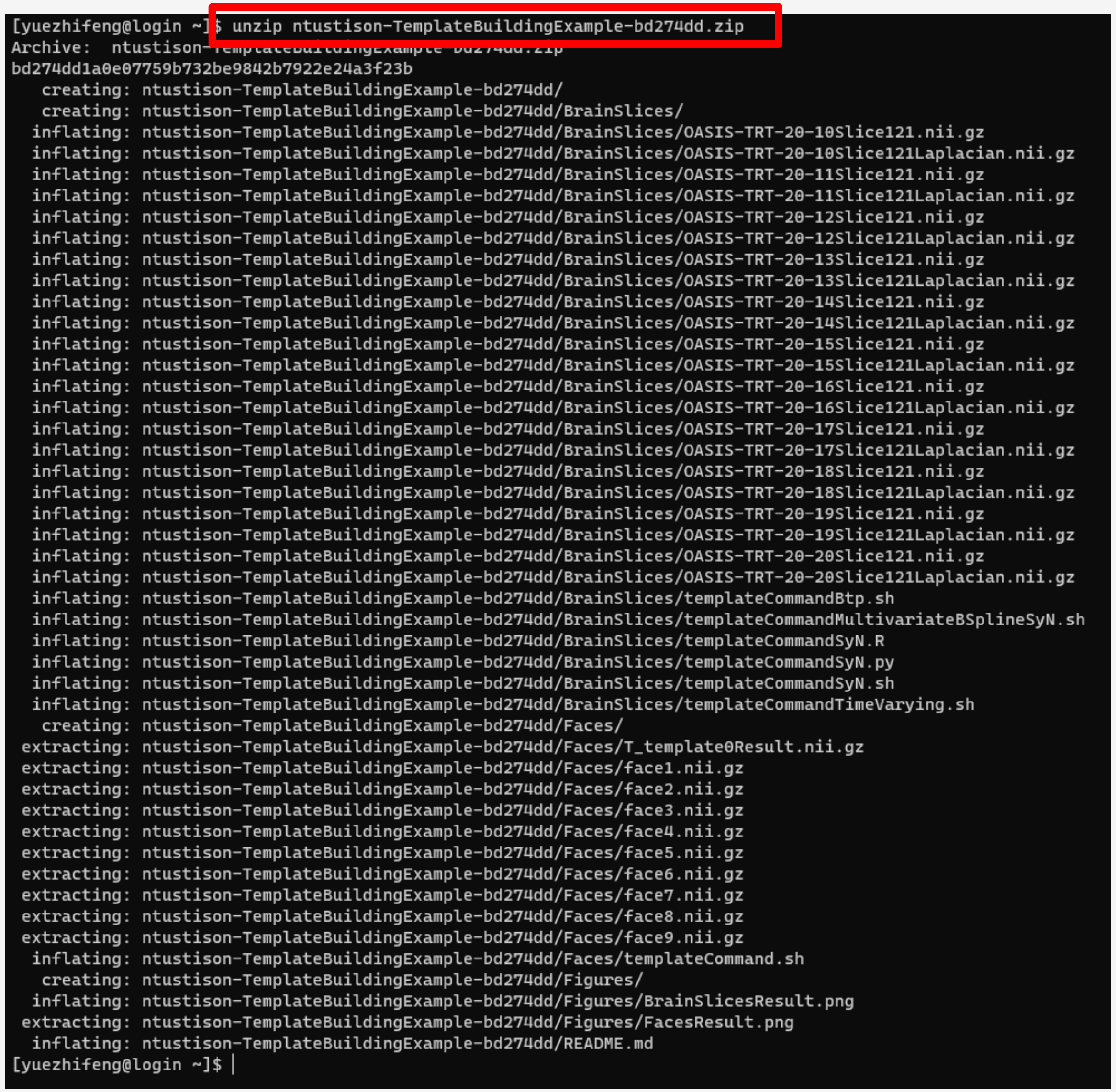
Step 3. ssh登录集群login节点



```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$ 
```

学习使用ants的singularity镜像制作平均脸

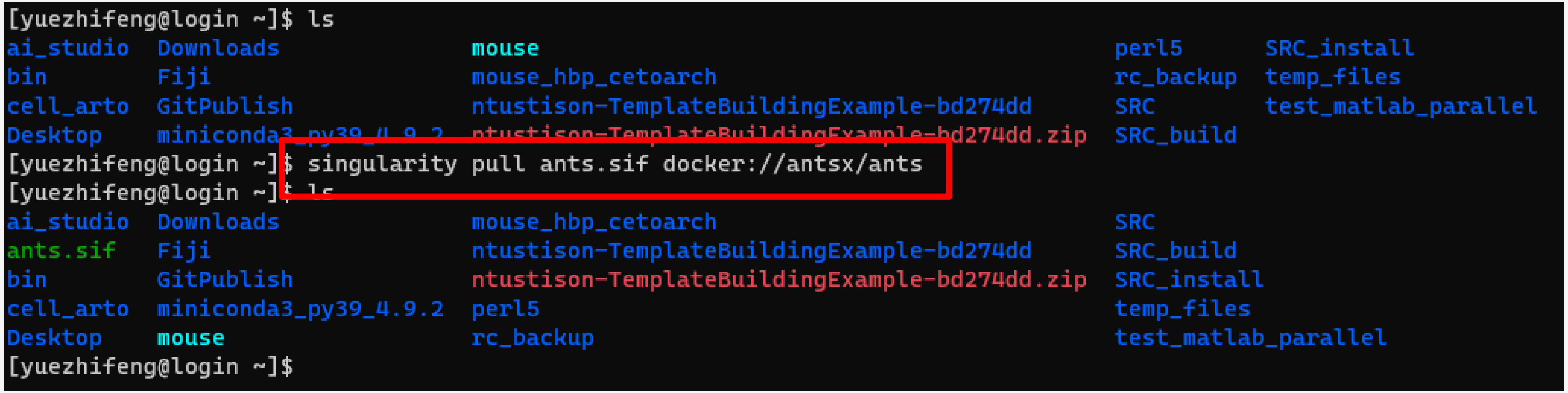
Step 4. 在login节点解压数据



```
[yuezhifeng@login ~]$ unzip ntustison-TemplateBuildingExample-bd274dd.zip
Archive:  ntustison-TemplateBuildingExample-bd274dd.zip
bd274dd1a0e07759b732be9842b7922e24a3f23b
   creating: ntustison-TemplateBuildingExample-bd274dd/
   creating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-10Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-10Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-11Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-11Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-12Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-12Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-13Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-13Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-14Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-14Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-15Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-15Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-16Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-16Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-17Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-17Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-18Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-18Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-19Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-19Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-20Slice121.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/OASIS-TRT-20-20Slice121Laplacian.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandBtp.sh
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandMultivariateBSplineSyN.sh
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandSyN.R
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandSyN.py
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandSyN.sh
  inflating: ntustison-TemplateBuildingExample-bd274dd/BrainSlices/templateCommandTimeVarying.sh
   creating: ntustison-TemplateBuildingExample-bd274dd/Faces/
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/T_template0Result.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face1.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face2.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face3.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face4.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face5.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face6.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face7.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face8.nii.gz
 extracting: ntustison-TemplateBuildingExample-bd274dd/Faces/face9.nii.gz
  inflating: ntustison-TemplateBuildingExample-bd274dd/Faces/templateCommand.sh
   creating: ntustison-TemplateBuildingExample-bd274dd/Figures/
  inflating: ntustison-TemplateBuildingExample-bd274dd/Figures/BrainSlicesResult.png
 extracting: ntustison-TemplateBuildingExample-bd274dd/Figures/FacesResult.png
  inflating: ntustison-TemplateBuildingExample-bd274dd/README.md
[yuezhifeng@login ~]$ 
```

学习使用ants的singularity镜像制作平均脸

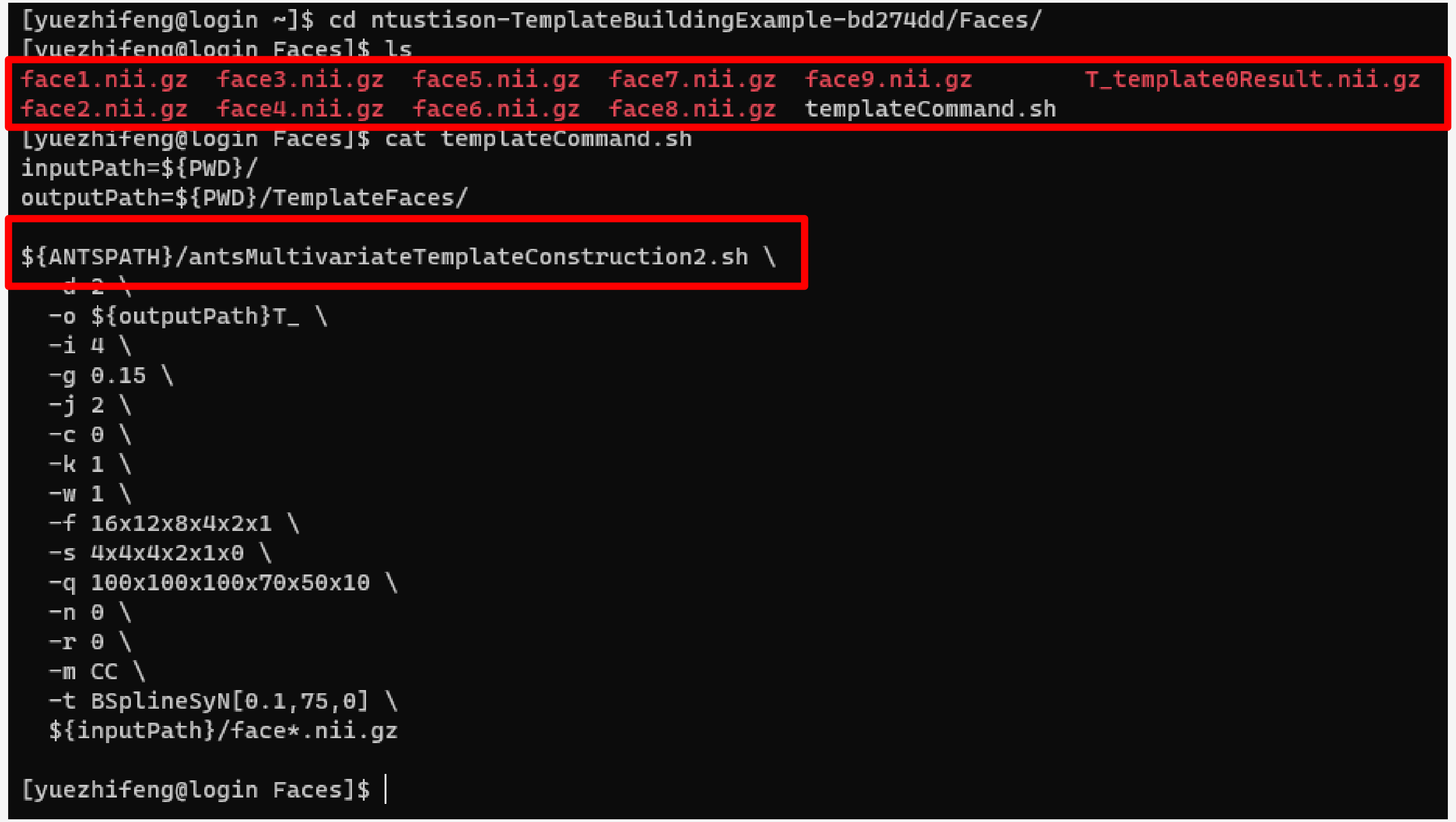
Step 5. 拉取ants的docker镜像，并保存成了singularity支持的sif格式镜像



```
[yuezhifeng@login ~]$ ls
ai_studio  Downloads              mouse                                          perl5      SRC_install
bin        Fiji                   mouse_hbp_cetoarch                             rc_backup  temp_files
cell_arto  GitPublish             ntustison-TemplateBuildingExample-bd274dd      SRC        test_matlab_parallel
Desktop    miniconda3_py39_4.9.2  ntustison-TemplateBuildingExample-bd274dd.zip  SRC_build
[yuezhifeng@login ~]$ singularity pull ants.sif docker://antsx/ants
[yuezhifeng@login ~]$ ls
ai_studio  Downloads              mouse_hbp_cetoarch                             SRC
ants.sif   Fiji                   ntustison-TemplateBuildingExample-bd274dd      SRC_build
bin        GitPublish             ntustison-TemplateBuildingExample-bd274dd.zip  SRC_install
cell_arto  miniconda3_py39_4.9.2  perl5                                          temp_files
Desktop    mouse                  rc_backup                                      test_matlab_parallel
[yuezhifeng@login ~]$
```

学习使用ants的singularity镜像制作平均脸

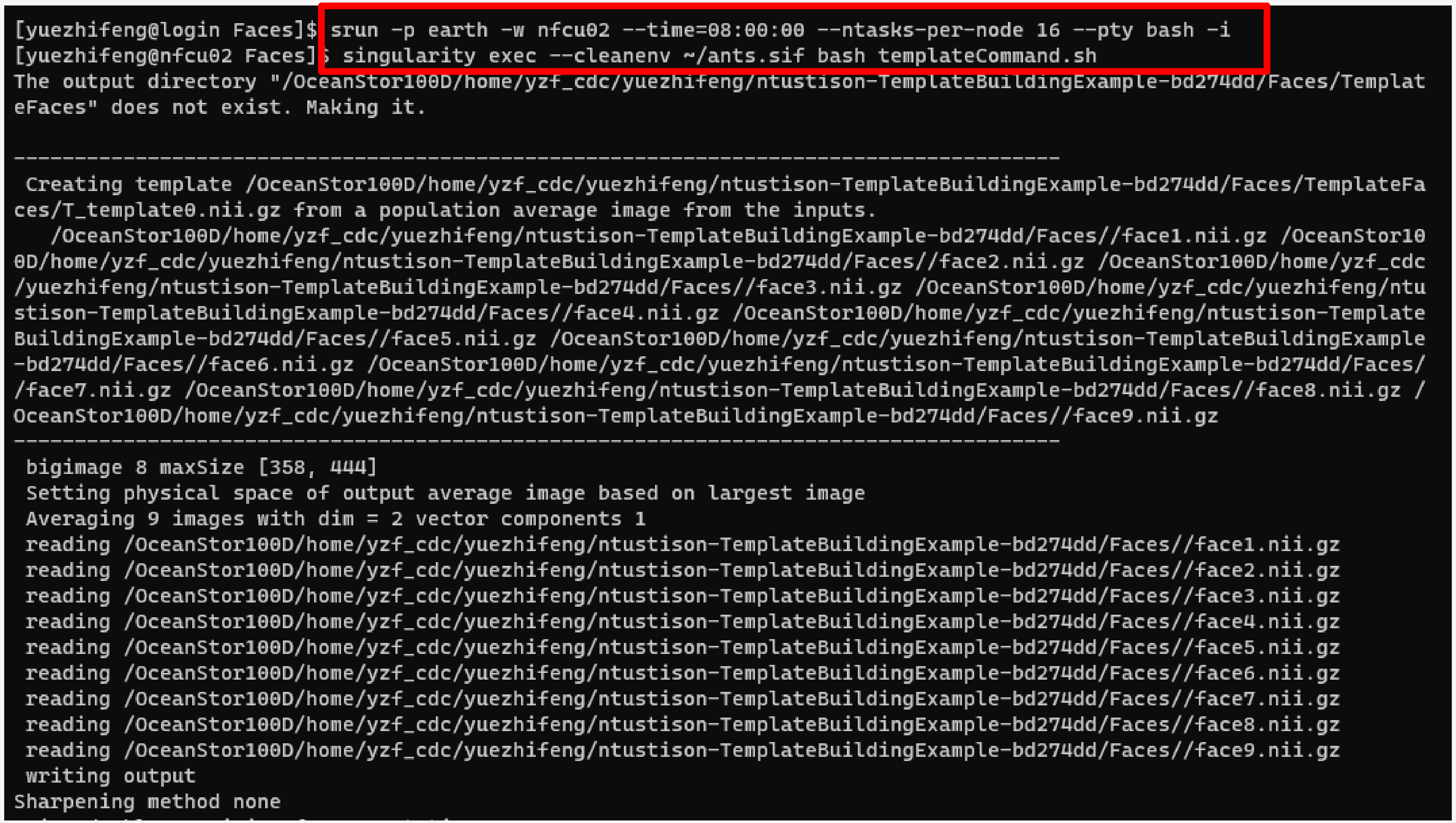
Step 6. 查看具体数据和运行的脚本



```
[yuezhifeng@login ~]$ cd ntustison-TemplateBuildingExample-bd274dd/Faces/
[yuezhifeng@login Faces]$ ls
face1.nii.gz  face3.nii.gz  face5.nii.gz  face7.nii.gz  face9.nii.gz         T_template0Result.nii.gz
face2.nii.gz  face4.nii.gz  face6.nii.gz  face8.nii.gz  templateCommand.sh
[yuezhifeng@login Faces]$ cat templateCommand.sh
inputPath=${PWD}/
outputPath=${PWD}/TemplateFaces/

${ANTSPATH}/antsMultivariateTemplateConstruction2.sh \
  -d 2 \
  -o ${outputPath}T_ \
  -i 4 \
  -g 0.15 \
  -j 2 \
  -c 0 \
  -k 1 \
  -w 1 \
  -f 16x12x8x4x2x1 \
  -s 4x4x4x2x1x0 \
  -q 100x100x100x70x50x10 \
  -n 0 \
  -r 0 \
  -m CC \
  -t BSplineSyN[0.1,75,0] \
  ${inputPath}/face*.nii.gz

[yuezhifeng@login Faces]$ 
```

学习使用ants的singularity镜像制作平均脸

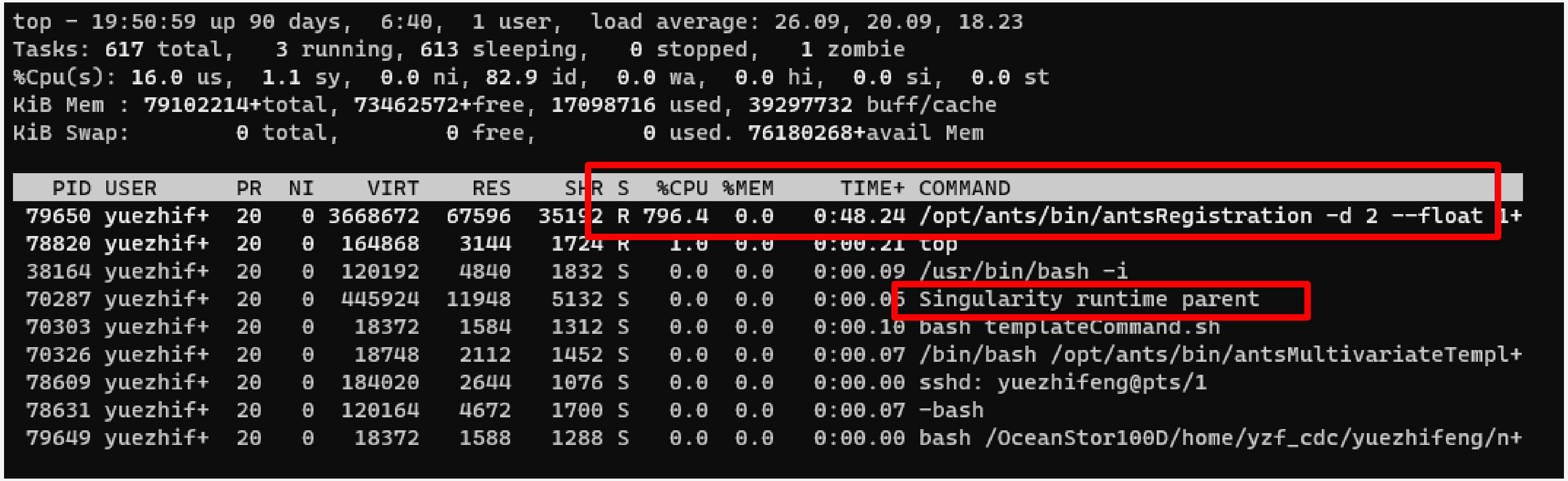
Step 7. 交互式的方式运行singularity容器中的ants模板制作工具链



```
[yuezhifeng@login Faces]$ srun -p earth -w nfcu02 --time=08:00:00 --ntasks-per-node 16 --pty bash -i
[yuezhifeng@nfcu02 Faces]$ singularity exec --cleanenv ~/ants.sif bash templateCommand.sh
The output directory "/OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/Templat
eFaces" does not exist. Making it.

--------------------------------------------------------------------------------------
 Creating template /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFa
ces/T_template0.nii.gz from a population average image from the inputs.
   /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face1.nii.gz /OceanStor10
0D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face2.nii.gz /OceanStor100D/home/yzf_cdc
/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face3.nii.gz /OceanStor100D/home/yzf_cdc/yuezhifeng/ntu
stison-TemplateBuildingExample-bd274dd/Faces//face4.nii.gz /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-Template
BuildingExample-bd274dd/Faces//face5.nii.gz /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample
-bd274dd/Faces//face6.nii.gz /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/
/face7.nii.gz /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face8.nii.gz /
OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face9.nii.gz
--------------------------------------------------------------------------------------
 bigimage 8 maxSize [358, 444]
 Setting physical space of output average image based on largest image
 Averaging 9 images with dim = 2 vector components 1
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face1.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face2.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face3.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face4.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face5.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face6.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face7.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face8.nii.gz
 reading /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces//face9.nii.gz
 writing output
Sharpening method none
```

学习使用ants的singularity镜像制作平均脸

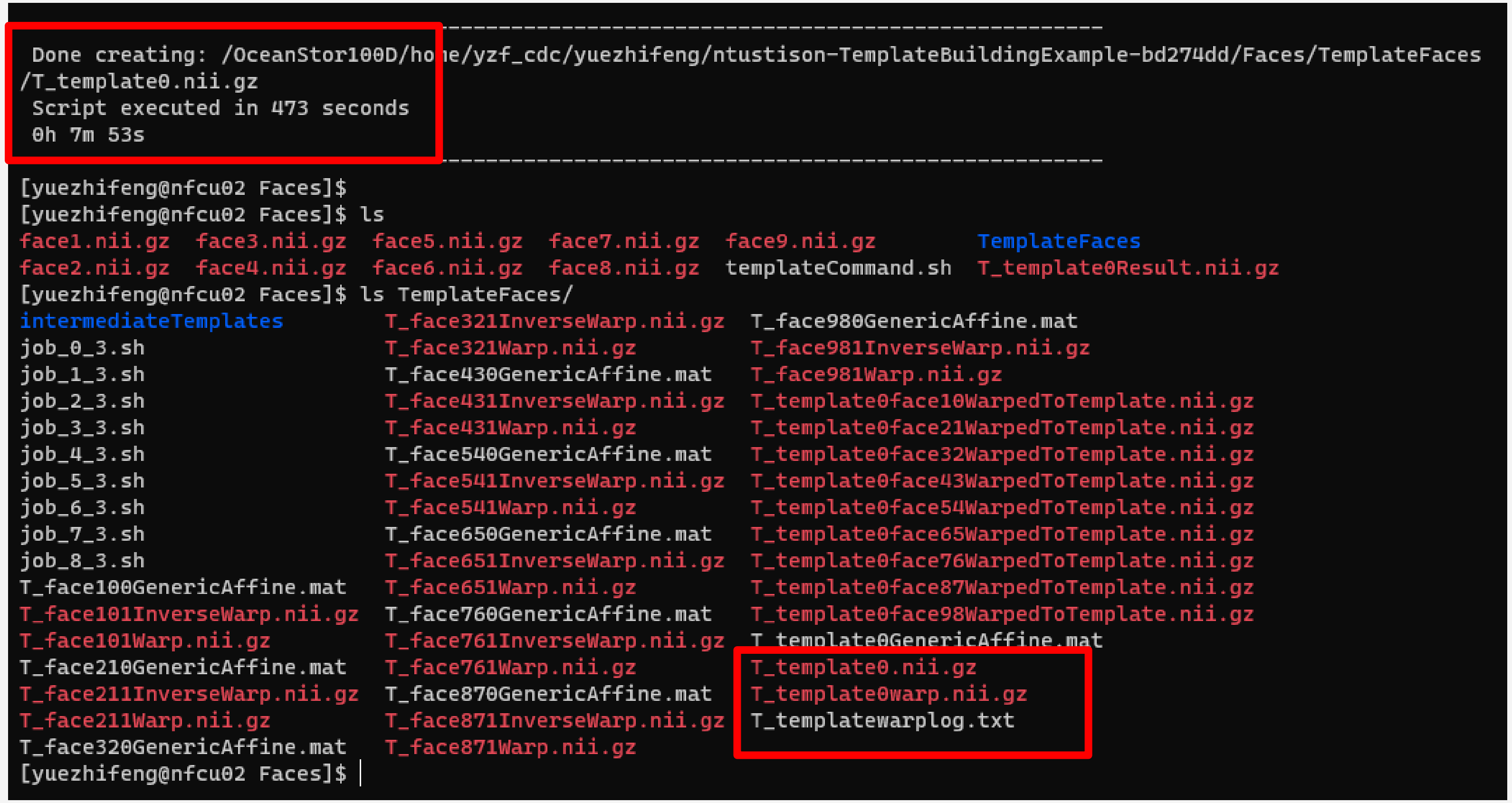
Step 8. 在计算过程查看运算是否正常



```
top - 19:50:59 up 90 days,  6:40,  1 user,  load average: 26.09, 20.09, 18.23
Tasks: 617 total,   3 running, 613 sleeping,   0 stopped,   1 zombie
%Cpu(s): 16.0 us,  1.1 sy,  0.0 ni, 82.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 79102214+total, 73462572+free, 17098716 used, 39297732 buff/cache
KiB Swap:        0 total,        0 free,        0 used. 76180268+avail Mem

   PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 79650 yuezhif+  20   0 3668672  67596  35192 R 796.4  0.0   0:48.24 /opt/ants/bin/antsRegistration -d 2 --float 1+
 78820 yuezhif+  20   0  164868   3144   1724 R   1.0  0.0   0:00.21 top
 38164 yuezhif+  20   0  120192   4840   1832 S   0.0  0.0   0:00.09 /usr/bin/bash -i
 70287 yuezhif+  20   0  445924  11948   5132 S   0.0  0.0   0:00.05 Singularity runtime parent
 70303 yuezhif+  20   0   18372   1584   1312 S   0.0  0.0   0:00.10 bash templateCommand.sh
 70326 yuezhif+  20   0   18748   2112   1452 S   0.0  0.0   0:00.07 /bin/bash /opt/ants/bin/antsMultivariateTempl+
 78609 yuezhif+  20   0  184020   2644   1076 S   0.0  0.0   0:00.00 sshd: yuezhifeng@pts/1
 78631 yuezhif+  20   0  120164   4672   1700 S   0.0  0.0   0:00.07 -bash
 79649 yuezhif+  20   0   18372   1588   1288 S   0.0  0.0   0:00.00 bash /OceanStor100D/home/yzf_cdc/yuezhifeng/n+
```

学习使用ants的singularity镜像制作平均脸

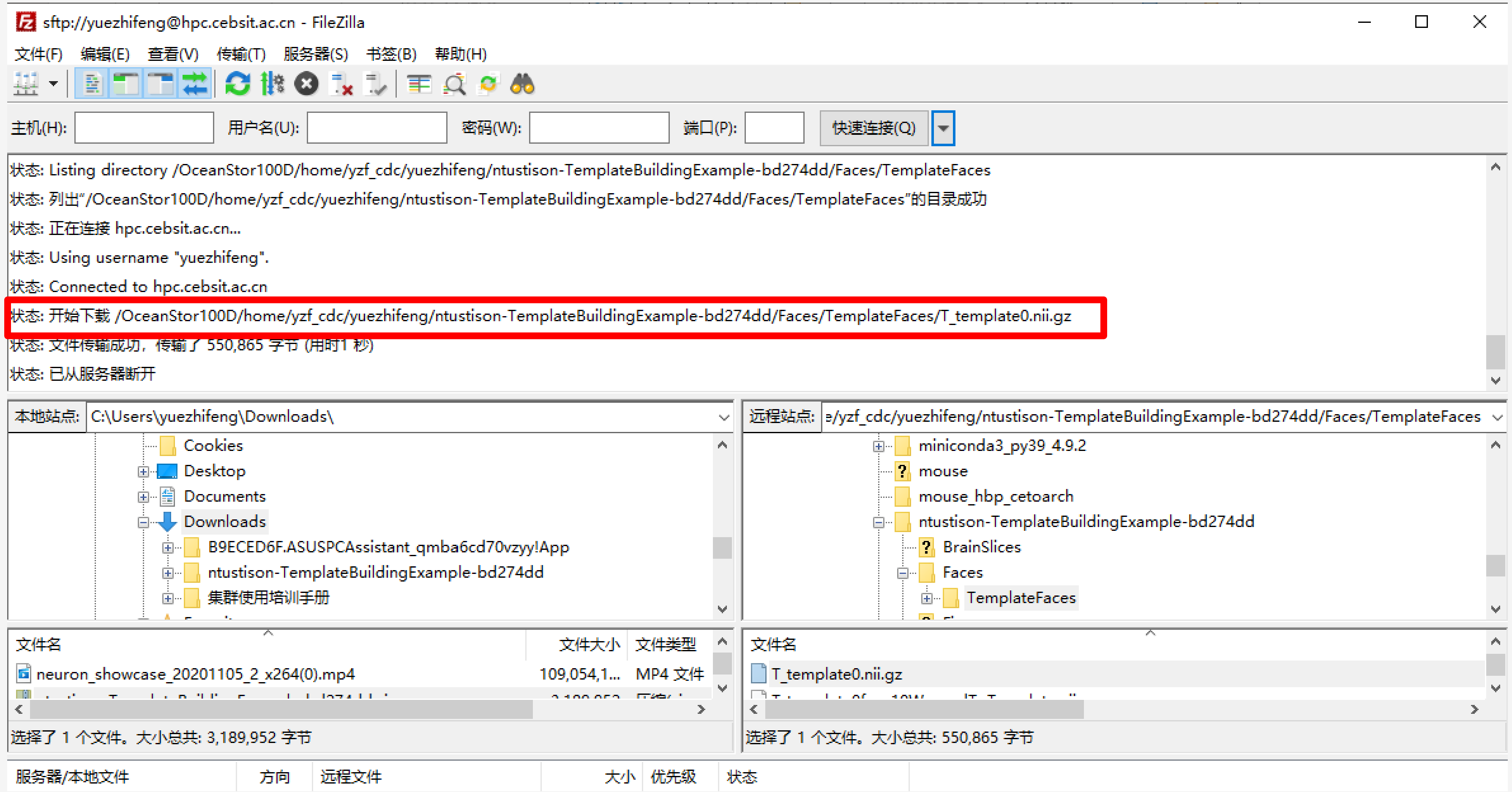
Step 9. 计算结束，查看生成的文件



```
--------------------------------------------------------------------------------------
 Done creating: /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFaces
/T_template0.nii.gz
 Script executed in 473 seconds
 0h 7m 53s
--------------------------------------------------------------------------------------
[yuezhifeng@nfcu02 Faces]$
[yuezhifeng@nfcu02 Faces]$ ls
face1.nii.gz  face3.nii.gz  face5.nii.gz  face7.nii.gz  face9.nii.gz        TemplateFaces
face2.nii.gz  face4.nii.gz  face6.nii.gz  face8.nii.gz  templateCommand.sh  T_template0Result.nii.gz
[yuezhifeng@nfcu02 Faces]$ ls TemplateFaces/
intermediateTemplates        T_face321InverseWarp.nii.gz  T_face980GenericAffine.mat
job_0_3.sh                   T_face321Warp.nii.gz         T_face981InverseWarp.nii.gz
job_1_3.sh                   T_face430GenericAffine.mat   T_face981Warp.nii.gz
job_2_3.sh                   T_face431InverseWarp.nii.gz  T_template0face10WarpedToTemplate.nii.gz
job_3_3.sh                   T_face431Warp.nii.gz         T_template0face21WarpedToTemplate.nii.gz
job_4_3.sh                   T_face540GenericAffine.mat   T_template0face32WarpedToTemplate.nii.gz
job_5_3.sh                   T_face541InverseWarp.nii.gz  T_template0face43WarpedToTemplate.nii.gz
job_6_3.sh                   T_face541Warp.nii.gz         T_template0face54WarpedToTemplate.nii.gz
job_7_3.sh                   T_face650GenericAffine.mat   T_template0face65WarpedToTemplate.nii.gz
job_8_3.sh                   T_face651InverseWarp.nii.gz  T_template0face76WarpedToTemplate.nii.gz
T_face100GenericAffine.mat   T_face651Warp.nii.gz         T_template0face87WarpedToTemplate.nii.gz
T_face101InverseWarp.nii.gz  T_face760GenericAffine.mat   T_template0face98WarpedToTemplate.nii.gz
T_face101Warp.nii.gz         T_face761InverseWarp.nii.gz  T_template0GenericAffine.mat
T_face210GenericAffine.mat   T_face761Warp.nii.gz         T_template0.nii.gz
T_face211InverseWarp.nii.gz  T_face870GenericAffine.mat   T_template0warp.nii.gz
T_face211Warp.nii.gz         T_face871InverseWarp.nii.gz  T_templatewarplog.txt
T_face320GenericAffine.mat   T_face871Warp.nii.gz
[yuezhifeng@nfcu02 Faces]$
```

学习使用ants的singularity镜像制作平均脸

Step 10. 下载制作的平均脸到个人计算机



sftp://yuezhifeng@hpc.cebsit.ac.cn - FileZilla

```
状态: Listing directory /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFaces
状态: 列出"/OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFaces"的目录成功
状态: 正在连接 hpc.cebsit.ac.cn...
状态: Using username "yuezhifeng".
状态: Connected to hpc.cebsit.ac.cn
状态: 开始下载 /OceanStor100D/home/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFaces/T_template0.nii.gz
状态: 文件传输成功，传输了 550,865 字节 (用时1 秒)
状态: 已从服务器断开
```

本地站点: C:\Users\yuezhifeng\Downloads\

Cookies
Desktop
Documents
Downloads
B9ECED6F.ASUSPCAssistant_qmba6cd70vzyy!App
ntustison-TemplateBuildingExample-bd274dd
集群使用培训手册

远程站点: e/yzf_cdc/yuezhifeng/ntustison-TemplateBuildingExample-bd274dd/Faces/TemplateFaces

文件名	文件大小	文件类型
neuron_showcase_20201105_2_x264(0).mp4	109,054,1...	MP4 文件

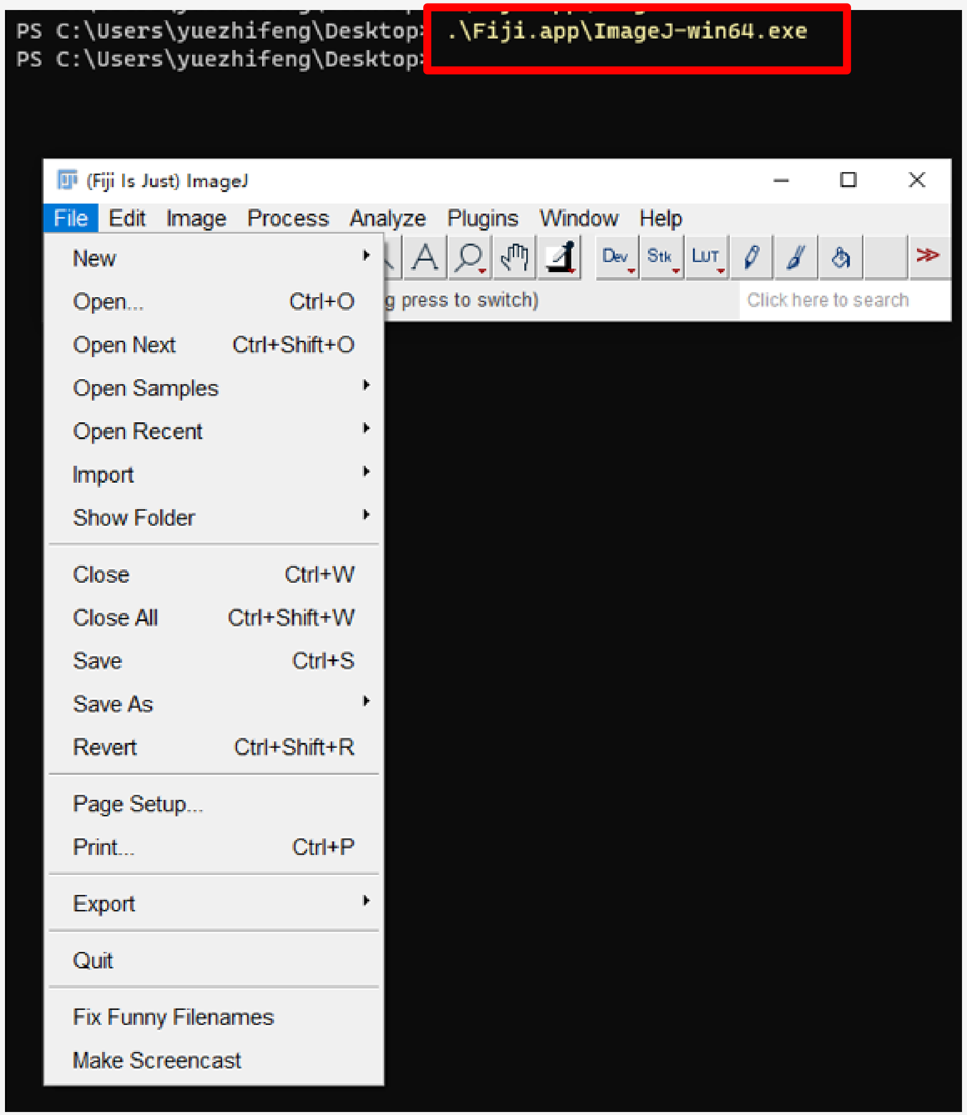
选择了 1 个文件。大小总共: 3,189,952 字节

文件名
T_template0.nii.gz

选择了 1 个文件。大小总共: 550,865 字节

学习使用ants的singularity镜像制作平均脸

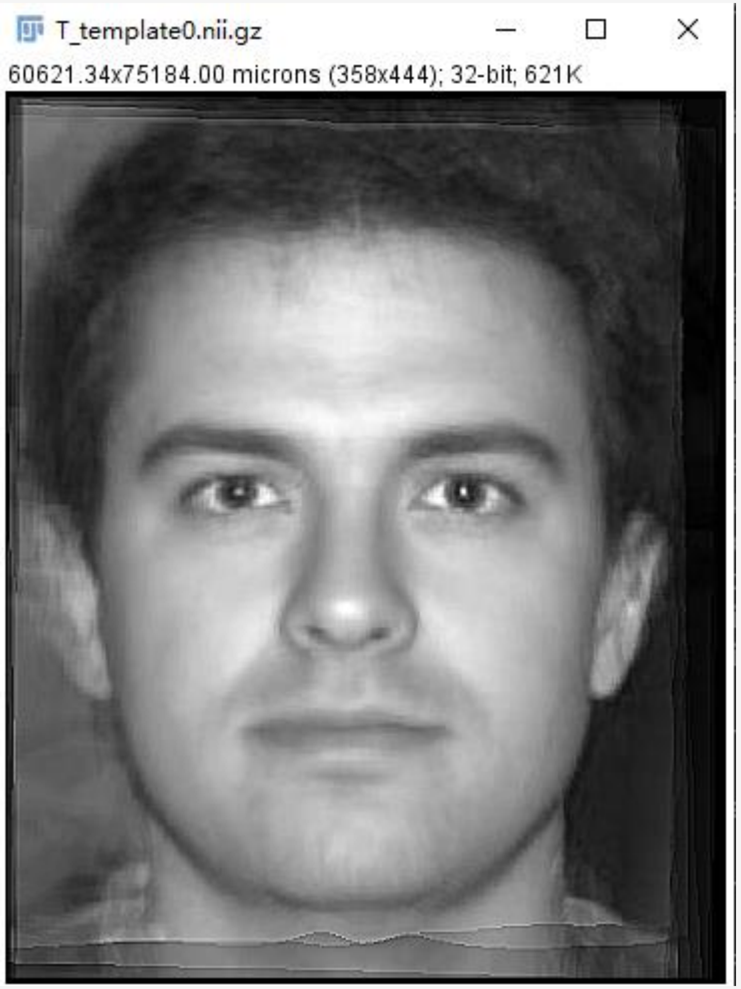
Step 11. 打开Fiji，用Fiji查看结果



```
PS C:\Users\yuezhifeng\Desktop> .\Fiji.app\ImageJ-win64.exe
PS C:\Users\yuezhifeng\Desktop>
```

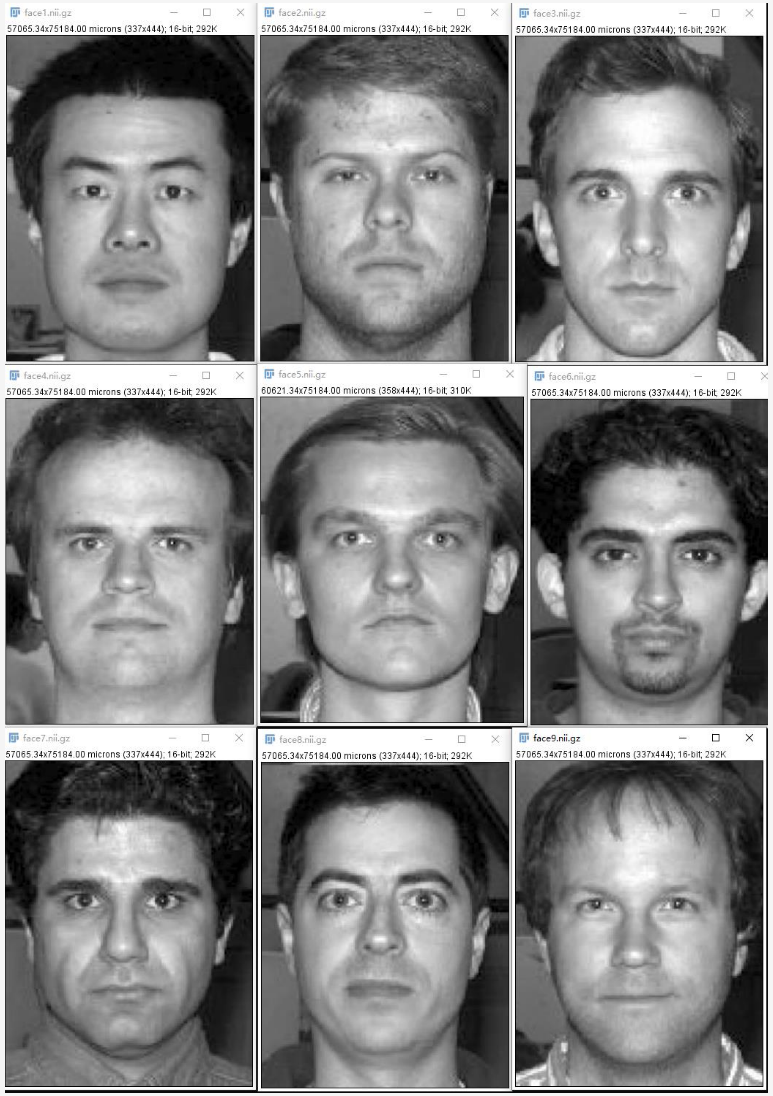
学习使用ants的singularity镜像制作平均脸

目标完成。
用右边9张脸制作的平均脸如下



T_template0.nii.gz

60621.34x75184.00 microns (358x444); 32-bit; 621K



face1.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face2.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face3.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face4.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face5.nii.gz

60621.34x75184.00 microns (358x444); 16-bit; 310K

face6.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face7.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face8.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

face9.nii.gz

57065.34x75184.00 microns (337x444); 16-bit; 292K

一 使用集群计算资源的方式

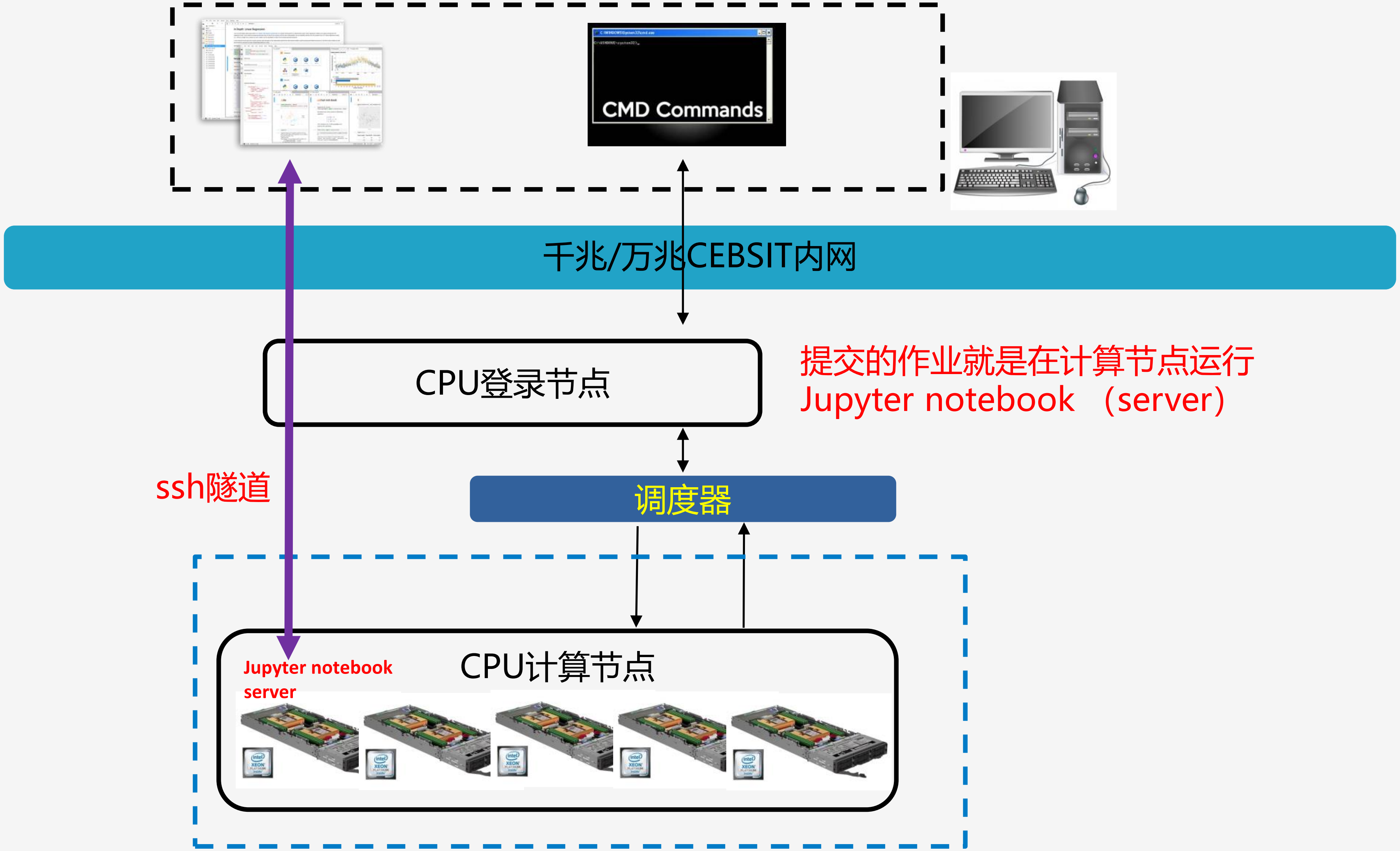
二 集群硬件和软件栈介绍

三 CPU集群容器工具介绍与容器示例

四 交互式使用计算资源的详细步骤

五 集群使用注意事项和账户申请流程

通过Slurm在计算节点启动Jupyter Notebook



千兆/万兆CEBSIT内网

CPU登录节点

提交的作业就是在计算节点运行
Jupyter notebook （server）

ssh隧道

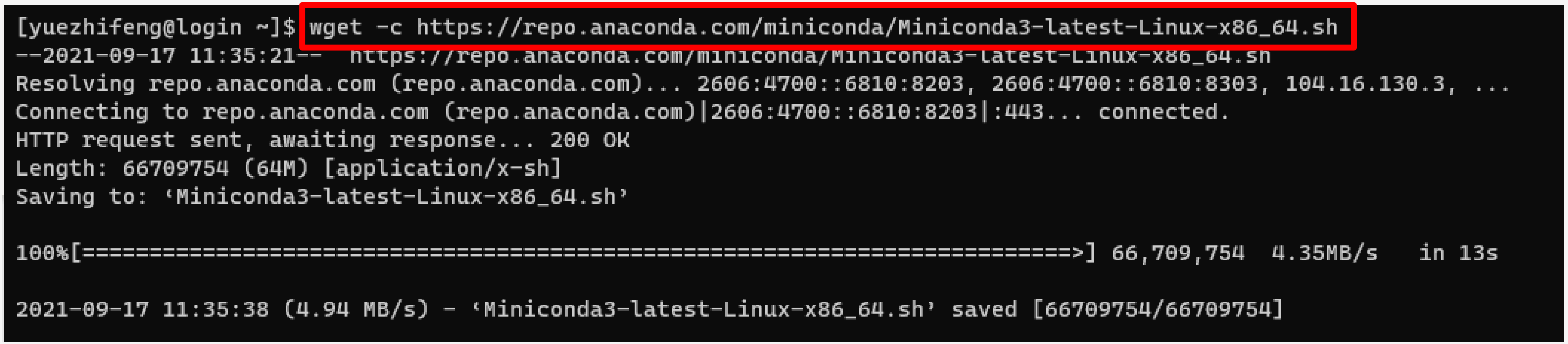
调度器

CPU计算节点

Jupyter notebook server

用conda安装jupyterlab和dask

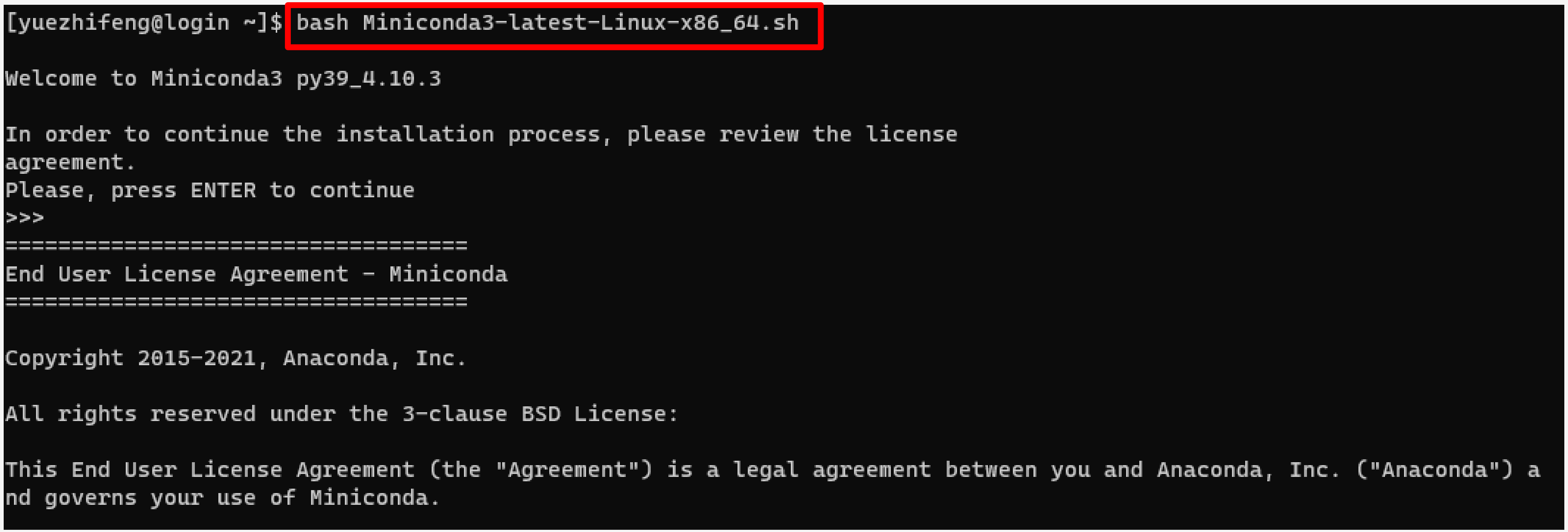
Step 2. 在login节点下载miniconda安装包



```
[yuezhifeng@login ~]$ wget -c https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
--2021-09-17 11:35:21--  https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 2606:4700::6810:8203, 2606:4700::6810:8303, 104.16.130.3, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|2606:4700::6810:8203|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 66709754 (64M) [application/x-sh]
Saving to: ‘Miniconda3-latest-Linux-x86_64.sh’

100%[==========================================================================>] 66,709,754  4.35MB/s   in 13s

2021-09-17 11:35:38 (4.94 MB/s) - ‘Miniconda3-latest-Linux-x86_64.sh’ saved [66709754/66709754]
```

用conda安装jupyterlab和dask

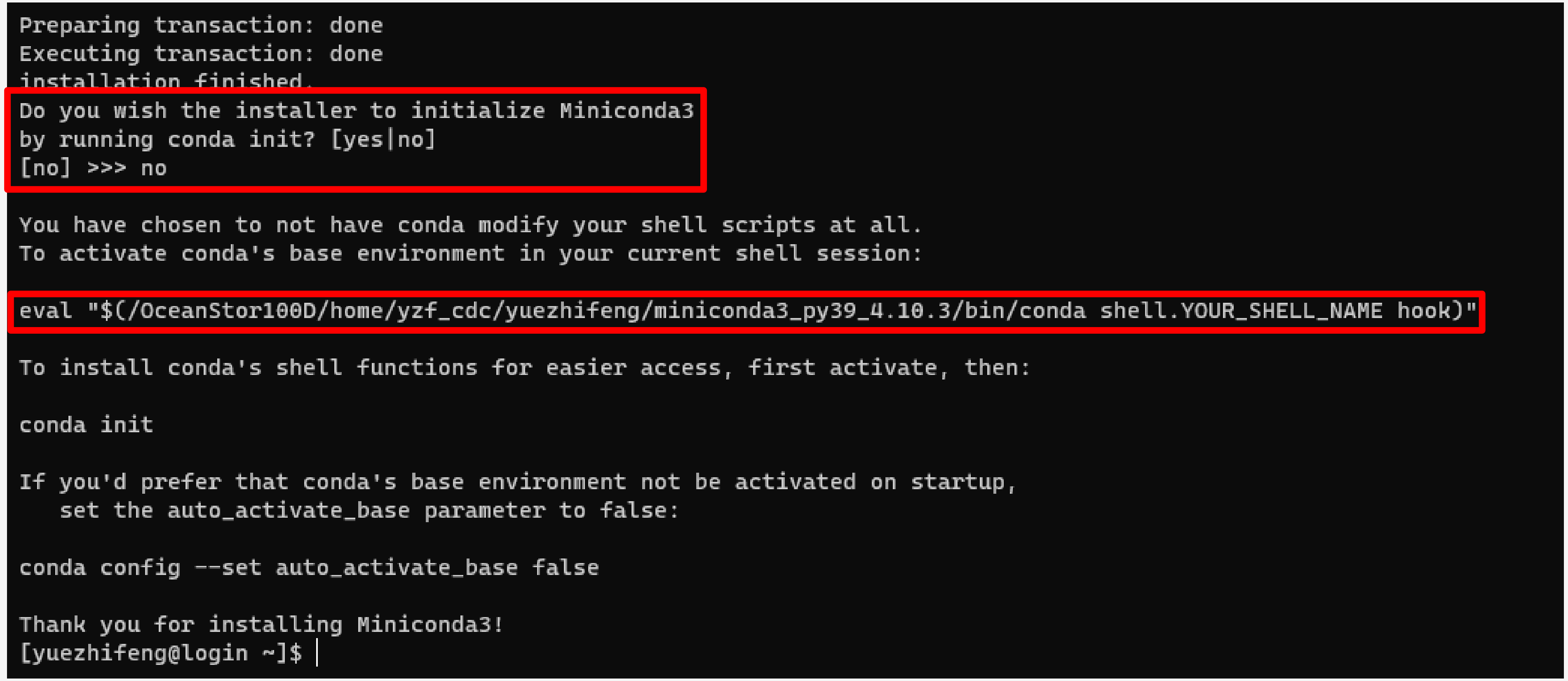
Step 3. 开始安装



```
[yuezhifeng@login ~]$ bash Miniconda3-latest-Linux-x86_64.sh

Welcome to Miniconda3 py39_4.10.3

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
===================================
End User License Agreement - Miniconda
===================================

Copyright 2015-2021, Anaconda, Inc.

All rights reserved under the 3-clause BSD License:

This End User License Agreement (the "Agreement") is a legal agreement between you and Anaconda, Inc. ("Anaconda") a
nd governs your use of Miniconda.
```

用conda安装jupyterlab和dask

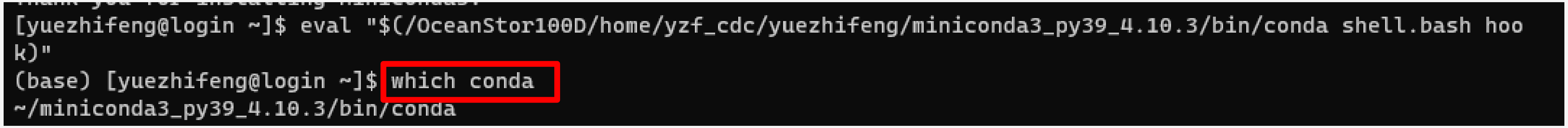
Step 4. 安装结束但不选择默认激活base



```
Preparing transaction: done
Executing transaction: done
installation finished.
Do you wish the installer to initialize Miniconda3
by running conda init? [yes|no]
[no] >>> no

You have chosen to not have conda modify your shell scripts at all.
To activate conda's base environment in your current shell session:

eval "$(/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/conda shell.YOUR_SHELL_NAME hook)"

To install conda's shell functions for easier access, first activate, then:

conda init

If you'd prefer that conda's base environment not be activated on startup,
   set the auto_activate_base parameter to false:

conda config --set auto_activate_base false

Thank you for installing Miniconda3!
[yuezhifeng@login ~]$
```

用conda安装jupyterlab和dask

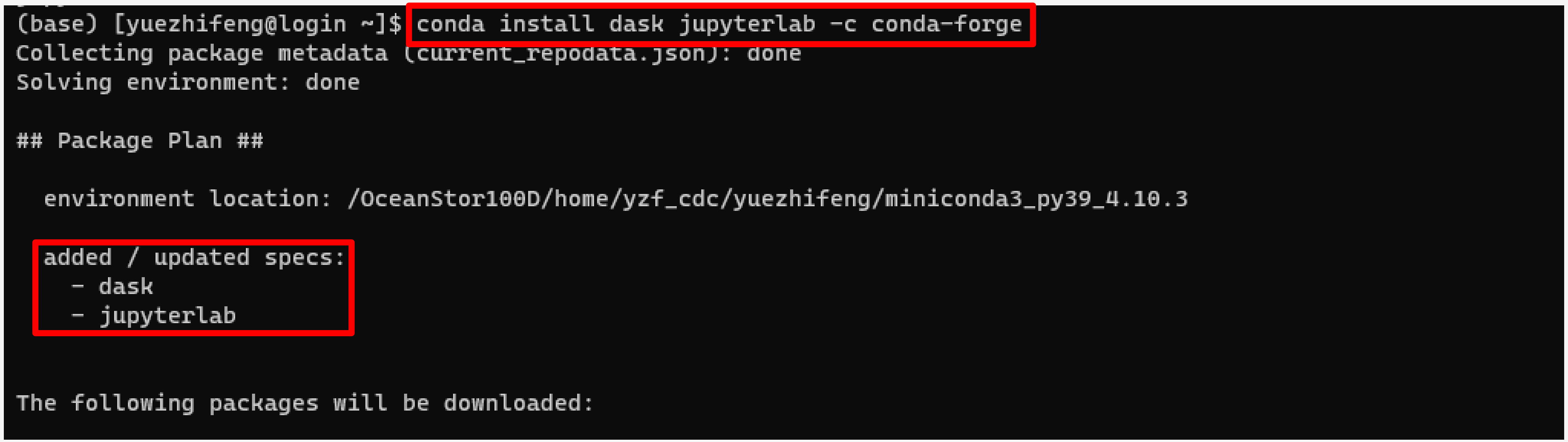
Step 5. 用如下命令每次想用conda再激活



```
Thank you for installing Miniconda3!
[yuezhifeng@login ~]$ eval "$(/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/conda shell.bash hoo
k)"
(base) [yuezhifeng@login ~]$ which conda
~/miniconda3_py39_4.10.3/bin/conda
```

用conda安装jupyterlab和dask

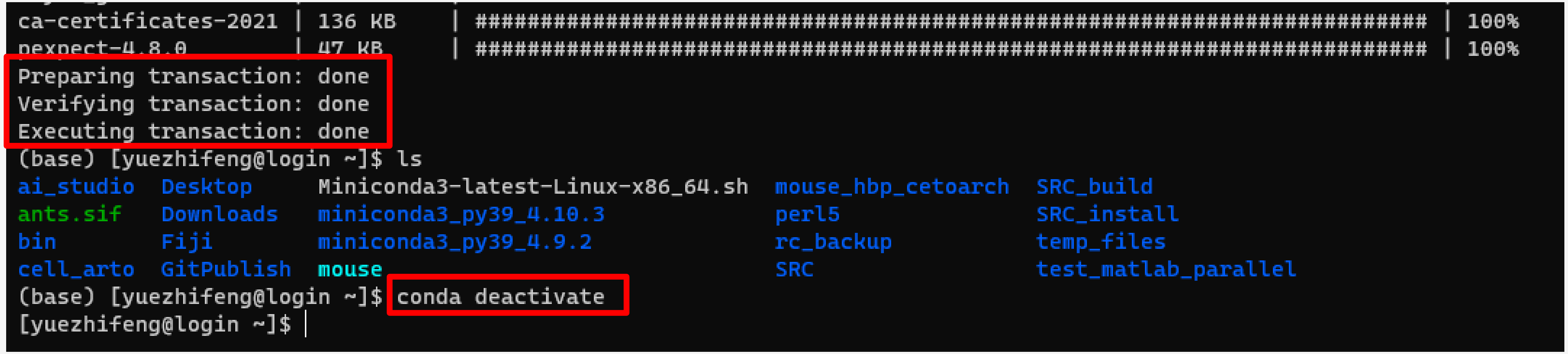
Step 6. 开始安装jupyterlab和dask



```
(base) [yuezhifeng@login ~]$ conda install dask jupyterlab -c conda-forge
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3

  added / updated specs:
    - dask
    - jupyterlab


The following packages will be downloaded:
```

用conda安装jupyterlab和dask

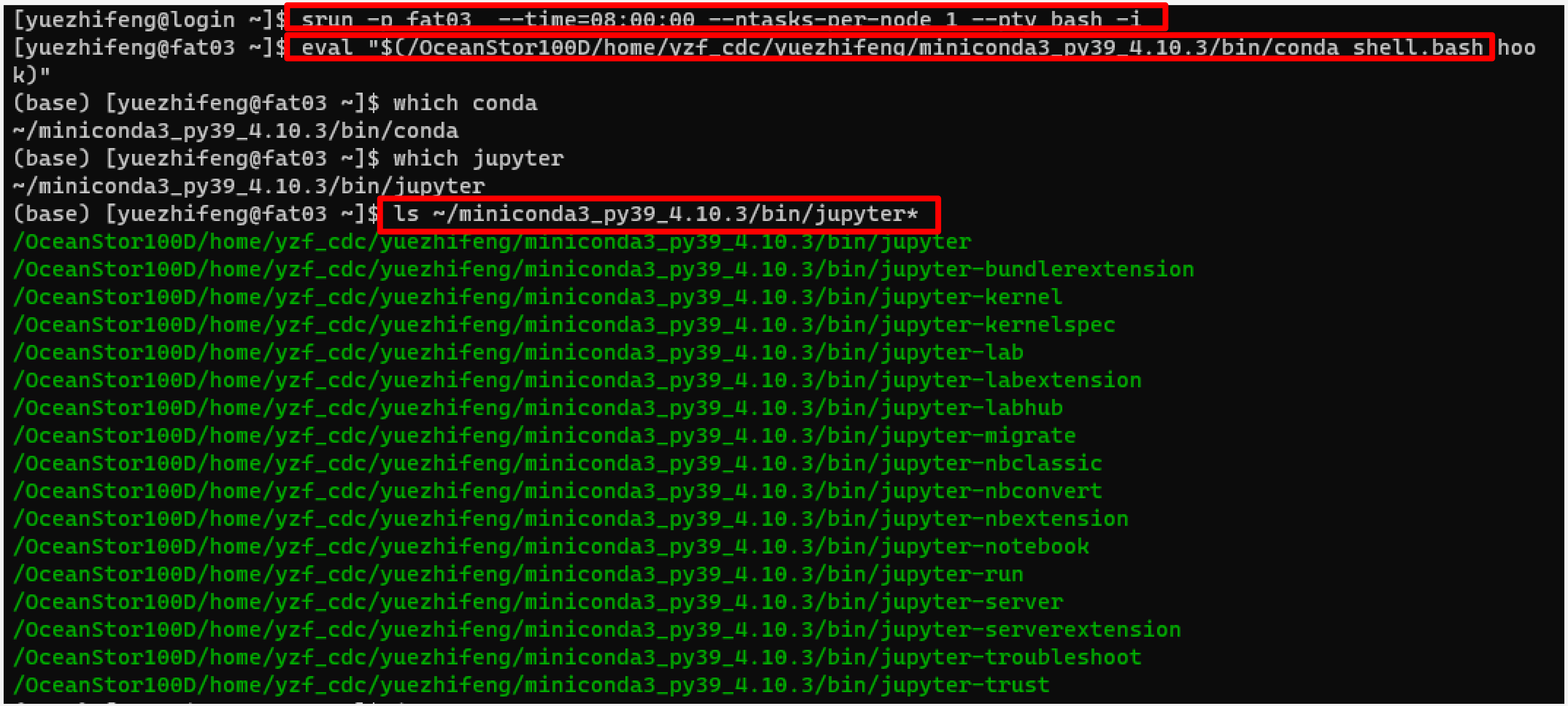
Step 7. 安装完成，退出conda环境



```
ca-certificates-2021 | 136 KB    | ######################################################################### | 100%
pexpect-4.8.0        | 47 KB     | ######################################################################### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
(base) [yuezhifeng@login ~]$ ls
ai_studio  Desktop     Miniconda3-latest-Linux-x86_64.sh  mouse_hbp_cetoarch  SRC_build
ants.sif   Downloads   miniconda3_py39_4.10.3             perl5               SRC_install
bin        Fiji        miniconda3_py39_4.9.2              rc_backup           temp_files
cell_arto  GitPublish  mouse                              SRC                 test_matlab_parallel
(base) [yuezhifeng@login ~]$ conda deactivate
[yuezhifeng@login ~]$
```

通过slurm拿资源

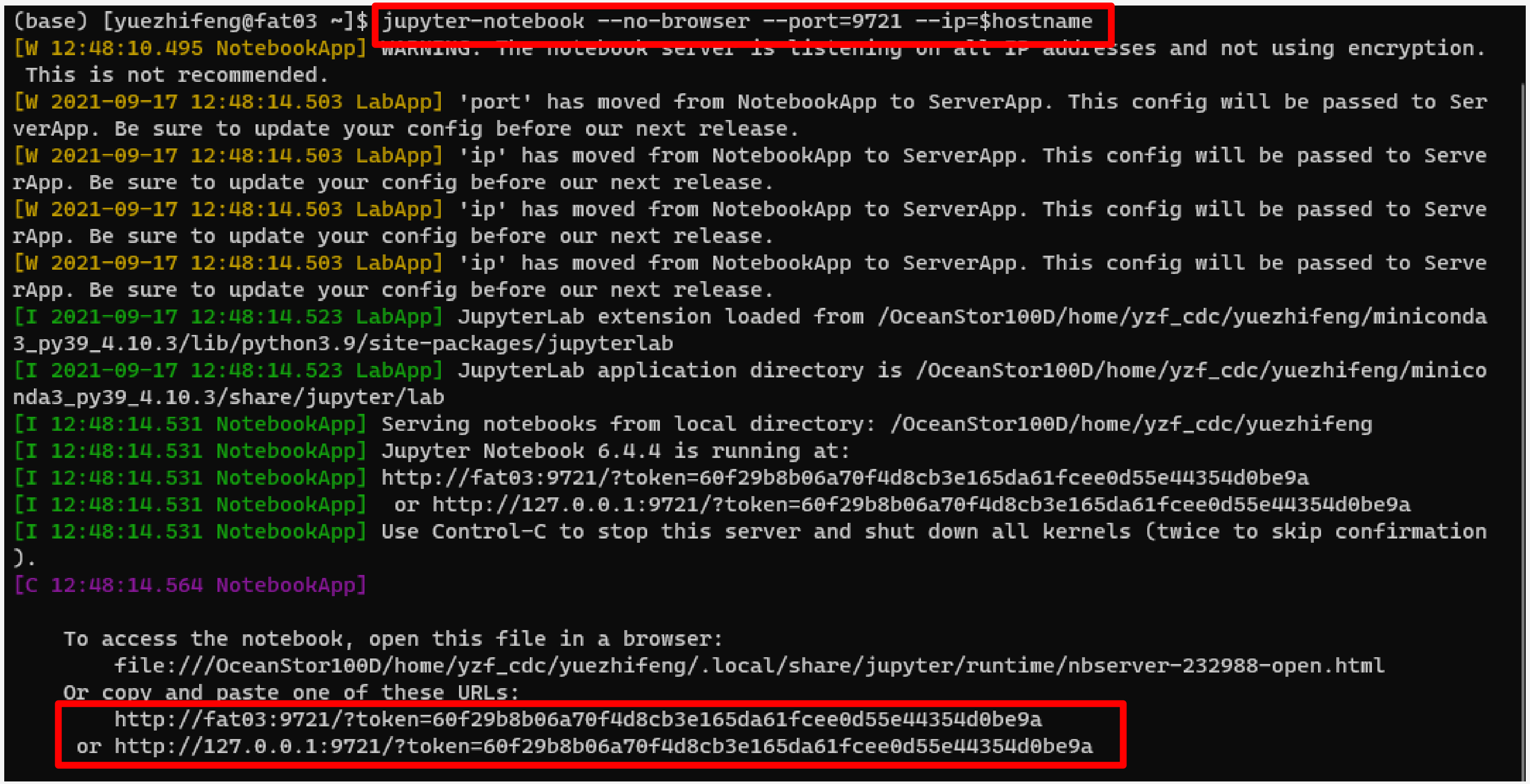
在计算节点查看Jupter相关的命令



```
[yuezhifeng@login ~]$ srun -p fat03  --time=08:00:00 --ntasks-per-node 1 --pty bash -i
[yuezhifeng@fat03 ~]$ eval "$(/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/conda shell.bash hoo
k)"
(base) [yuezhifeng@fat03 ~]$ which conda
~/miniconda3_py39_4.10.3/bin/conda
(base) [yuezhifeng@fat03 ~]$ which jupyter
~/miniconda3_py39_4.10.3/bin/jupyter
(base) [yuezhifeng@fat03 ~]$ ls ~/miniconda3_py39_4.10.3/bin/jupyter*
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-bundlerextension
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-kernel
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-kernelspec
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-lab
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-labextension
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-labhub
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-migrate
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-nbclassic
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-nbconvert
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-nbextension
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-notebook
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-run
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-server
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-serverextension
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-troubleshoot
/OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda3_py39_4.10.3/bin/jupyter-trust
```

交互式的方式启动jupyter notebook server

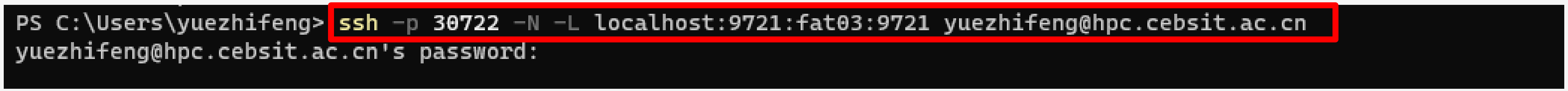
启动jupyter notebook server



```
(base) [yuezhifeng@fat03 ~]$ jupyter-notebook --no-browser --port=9721 --ip=$hostname
[W 12:48:10.495 NotebookApp] WARNING: The notebook server is listening on all IP addresses and not using encryption.
 This is not recommended.
[W 2021-09-17 12:48:14.503 LabApp] 'port' has moved from NotebookApp to ServerApp. This config will be passed to Ser
verApp. Be sure to update your config before our next release.
[W 2021-09-17 12:48:14.503 LabApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to Serve
rApp. Be sure to update your config before our next release.
[W 2021-09-17 12:48:14.503 LabApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to Serve
rApp. Be sure to update your config before our next release.
[W 2021-09-17 12:48:14.503 LabApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to Serve
rApp. Be sure to update your config before our next release.
[I 2021-09-17 12:48:14.523 LabApp] JupyterLab extension loaded from /OceanStor100D/home/yzf_cdc/yuezhifeng/miniconda
3_py39_4.10.3/lib/python3.9/site-packages/jupyterlab
[I 2021-09-17 12:48:14.523 LabApp] JupyterLab application directory is /OceanStor100D/home/yzf_cdc/yuezhifeng/minico
nda3_py39_4.10.3/share/jupyter/lab
[I 12:48:14.531 NotebookApp] Serving notebooks from local directory: /OceanStor100D/home/yzf_cdc/yuezhifeng
[I 12:48:14.531 NotebookApp] Jupyter Notebook 6.4.4 is running at:
[I 12:48:14.531 NotebookApp] http://fat03:9721/?token=60f29b8b06a70f4d8cb3e165da61fcee0d55e44354d0be9a
[I 12:48:14.531 NotebookApp]  or http://127.0.0.1:9721/?token=60f29b8b06a70f4d8cb3e165da61fcee0d55e44354d0be9a
[I 12:48:14.531 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation
).
[C 12:48:14.564 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///OceanStor100D/home/yzf_cdc/yuezhifeng/.local/share/jupyter/runtime/nbserver-232988-open.html
    Or copy and paste one of these URLs:
        http://fat03:9721/?token=60f29b8b06a70f4d8cb3e165da61fcee0d55e44354d0be9a
     or http://127.0.0.1:9721/?token=60f29b8b06a70f4d8cb3e165da61fcee0d55e44354d0be9a
```

在Jupyter中进行Dask编程

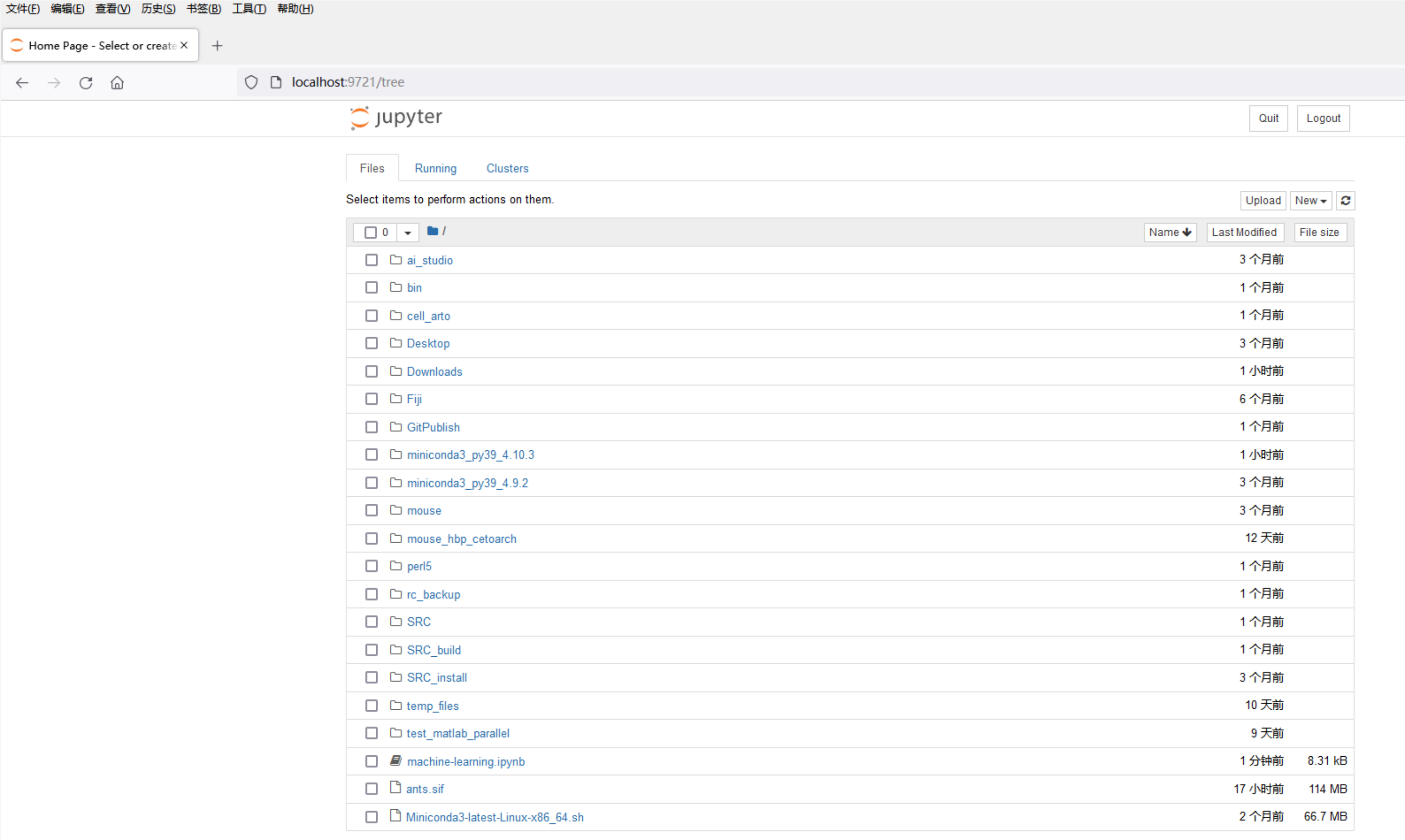
在本地机器建立ssh隧道



```
PS C:\Users\yuezhifeng> ssh -p 30722 -N -L localhost:9721:fat03:9721 yuezhifeng@hpc.cebsit.ac.cn
yuezhifeng@hpc.cebsit.ac.cn's password:
```

在Jupyter中进行Dask编程

连接成功

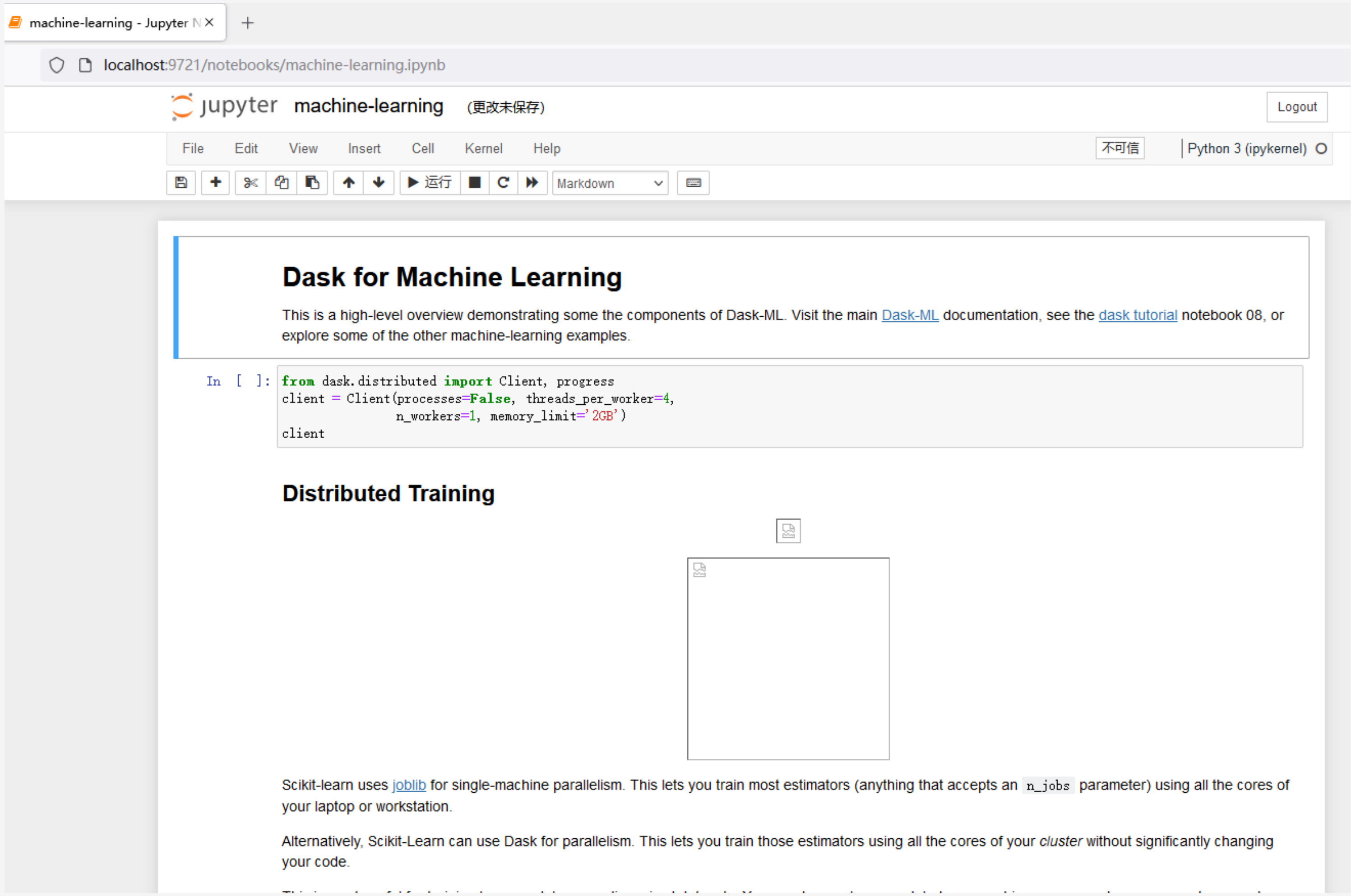

文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)
Home Page - Select or create
localhost:9721/tree
jupyter
Quit
Logout
Files
Running
Clusters
Upload
New
0
/

Select items to perform actions on them.

Name	Last Modified	File size
ai_studio	3 个月前	
bin	1 个月前	
cell_arto	1 个月前	
Desktop	3 个月前	
Downloads	1 小时前	
Fiji	6 个月前	
GitPublish	1 个月前	
miniconda3_py39_4.10.3	1 小时前	
miniconda3_py39_4.9.2	3 个月前	
mouse	3 个月前	
mouse_hbp_cetoarch	12 天前	
perl5	1 个月前	
rc_backup	1 个月前	
SRC	1 个月前	
SRC_build	1 个月前	
SRC_install	3 个月前	
temp_files	10 天前	
test_matlab_parallel	9 天前	
machine-learning.ipynb	1 分钟前	8.31 kB
ants.sif	17 小时前	114 MB
Miniconda3-latest-Linux-x86_64.sh	2 个月前	66.7 MB

在Jupyter中进行Dask编程

打开machine_learning.ipynb



Dask for Machine Learning

This is a high-level overview demonstrating some the components of Dask-ML. Visit the main Dask-ML documentation, see the dask tutorial notebook 08, or explore some of the other machine-learning examples.

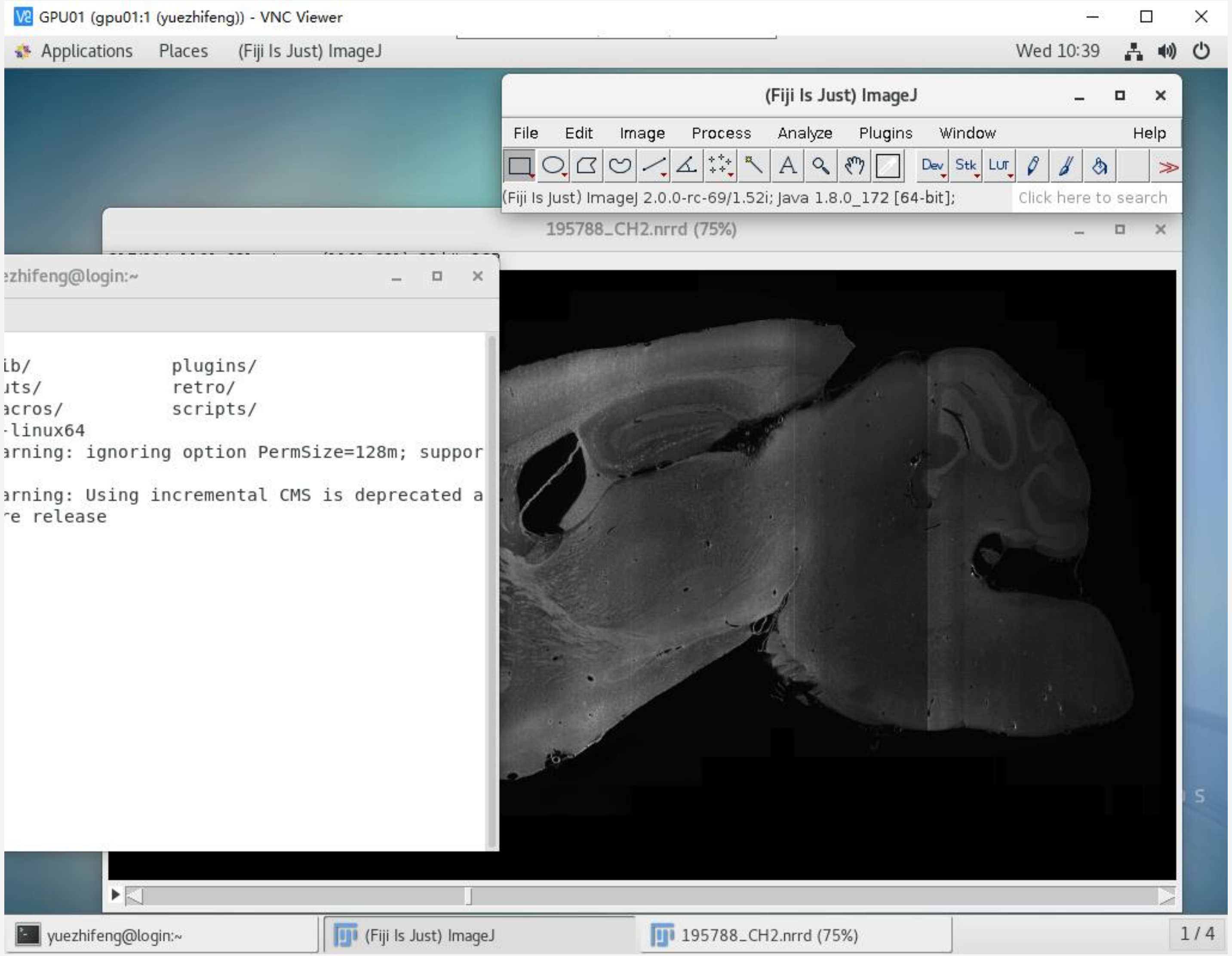
In [ ]:
```
from dask.distributed import Client, progress
client = Client(processes=False, threads_per_worker=4,
                n_workers=1, memory_limit='2GB')
client
```

Distributed Training

Scikit-learn uses joblib for single-machine parallelism. This lets you train most estimators (anything that accepts an `n_jobs` parameter) using all the cores of your laptop or workstation.

Alternatively, Scikit-Learn can use Dask for parallelism. This lets you train those estimators using all the cores of your *cluster* without significantly changing your code.

交互式使用计算节点的方式：3. VNC



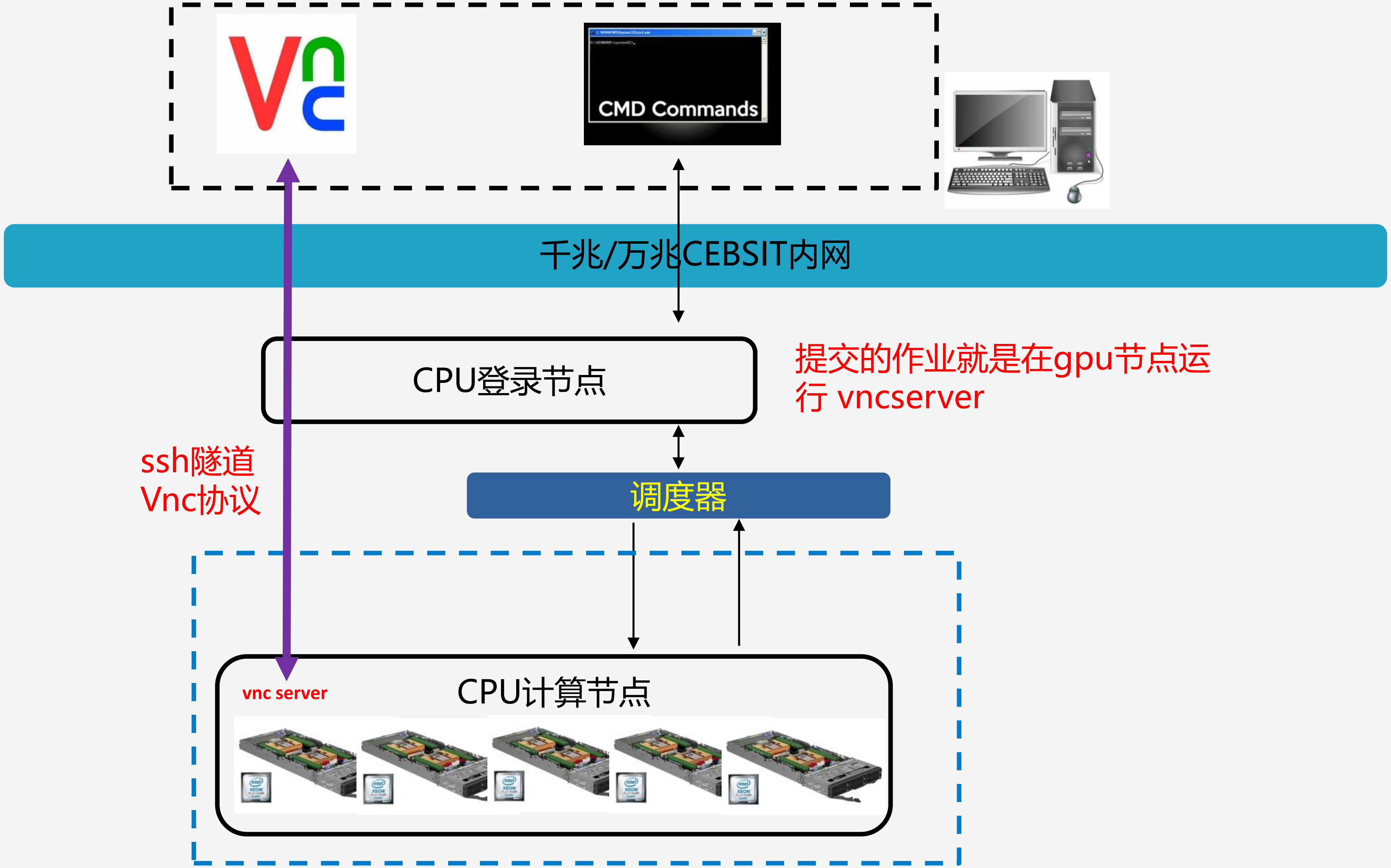
(Fiji Is Just) ImageJ 2.0.0-rc-69/1.52i; Java 1.8.0_172 [64-bit];

```
ib/              plugins/
uts/             retro/
acros/           scripts/
-linux64
arning: ignoring option PermSize=128m; suppor

arning: Using incremental CMS is deprecated a
re release
```

VNC (Virtual Network Console)，即虚拟网络控制台，它是一款基于 UNIX 和 Linux 操作系统的优秀远程控制工具软件，由著名的 AT&T 的欧洲研究实验室开发，远程控制能力强大，高效实用，并且免费开源。

VNC基本上是由两部分组成：一部分是客户端的应用程序(vncviewer)；另外一部分是服务器端的应用程序(vncserver)。在任何安装了客户端的应用程序(vncviewer)的计算机都能十分方便地与安装了服务器端的应用程序(vncserver)的计算机相互连接。

通过Slurm在gpu节点启动vnc服务



千兆/万兆CEBSIT内网

CPU登录节点

提交的作业就是在gpu节点运行 vncserver

ssh隧道
Vnc协议

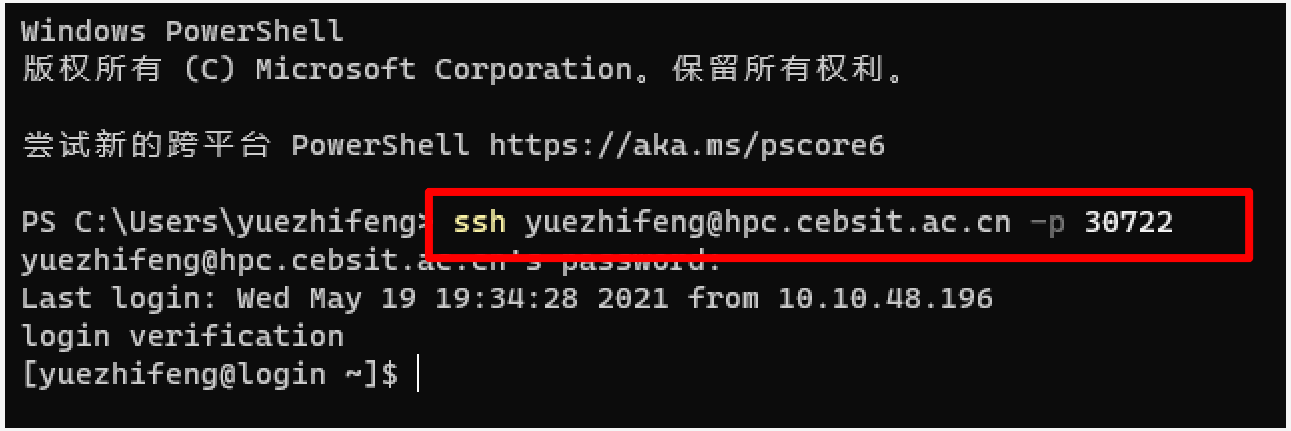
调度器

vnc server

CPU计算节点

VNC远程桌面：Step 1. ssh方式登录集群

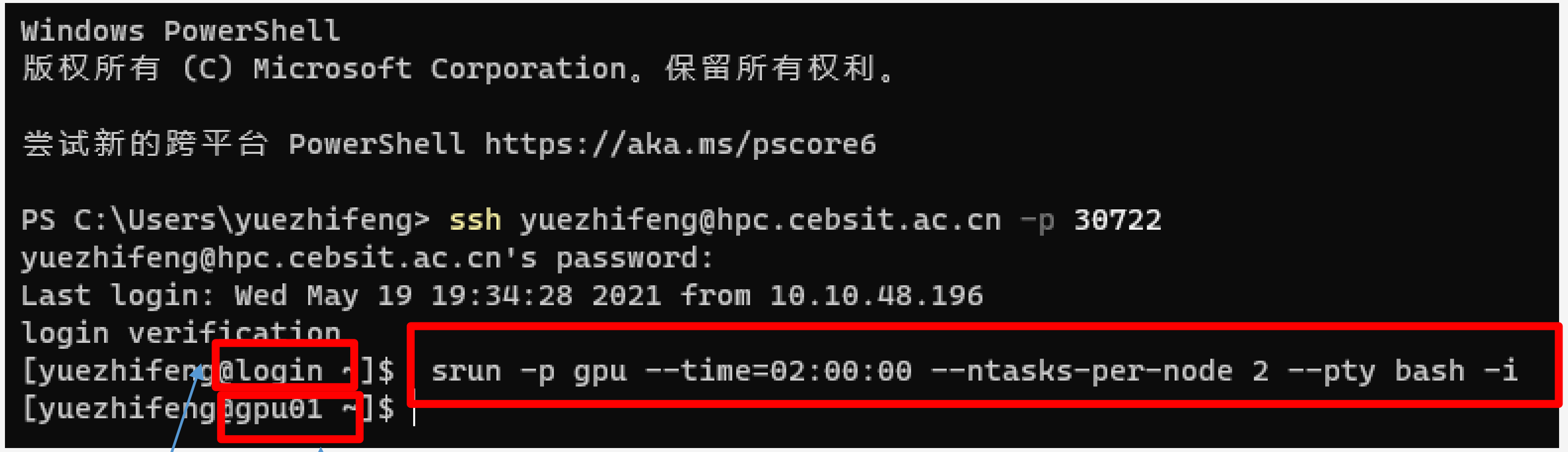
以powershell为例



```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$
```

VNC远程桌面：Step 2. 申请gpu01的资源

运行的命令为：srun -p gpu --time=02:00:00 --ntasks-per-node 2 --pty bash -i

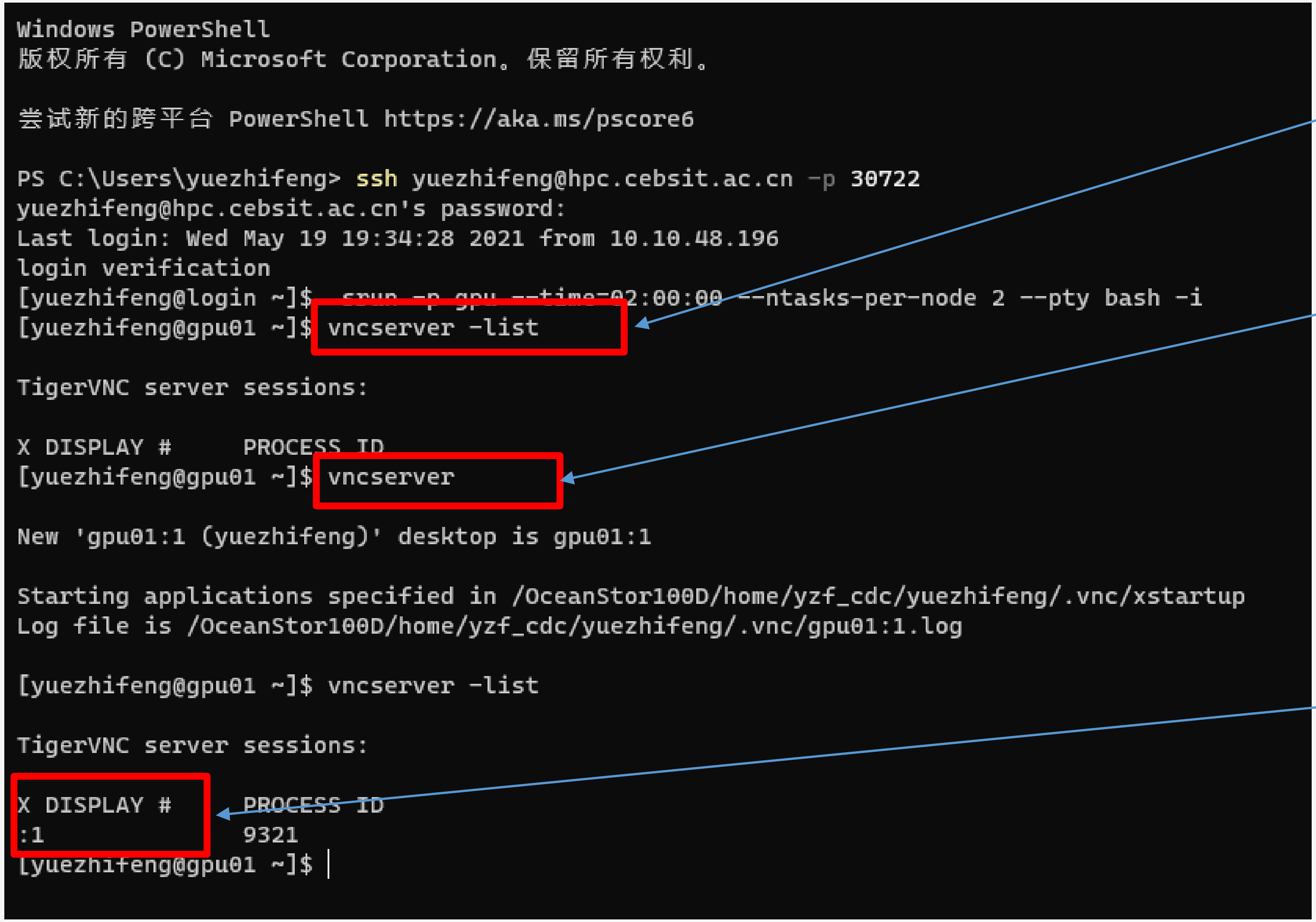


```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$  srun -p gpu --time=02:00:00 --ntasks-per-node 2 --pty bash -i
[yuezhifeng@gpu01 ~]$ 
```

已经跑到gpu01上了

VNC远程桌面：Step 3.启动vncserver服务

运行的命令为：vncserver



```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$  srun -p gpu --time=02:00:00 --ntasks-per-node 2 --pty bash -i
[yuezhifeng@gpu01 ~]$ vncserver -list

TigerVNC server sessions:

X DISPLAY #     PROCESS ID
[yuezhifeng@gpu01 ~]$ vncserver

New 'gpu01:1 (yuezhifeng)' desktop is gpu01:1

Starting applications specified in /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/xstartup
Log file is /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/gpu01:1.log

[yuezhifeng@gpu01 ~]$ vncserver -list

TigerVNC server sessions:

X DISPLAY #     PROCESS ID
:1              9321
[yuezhifeng@gpu01 ~]$
```

先看看有没有自己的vncserver服务

没有的话启动一个vncserver服务；首次运行vncserver前用vncpasswd设置一个密码，后面要用。

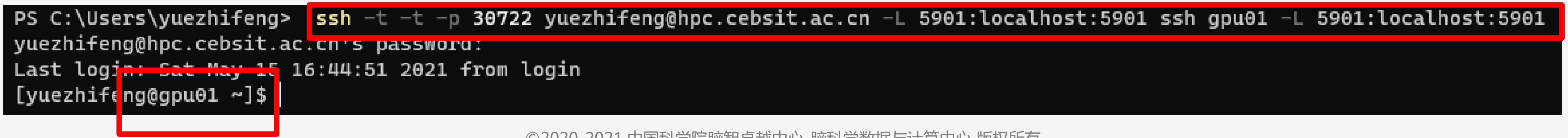
记住这个端口号 “X”

VNC远程桌面：Step 4. 转发vncserver服务

在新打开的powershell里面运行的命令为：ssh -t -t -p 30722 yuezhifeng@hpc.cebsit.ac.cn -L 5901:localhost:5901 ssh gpu01 -L 5901:localhost:5901

这里的端口号用5900加上step3中记下的“X”，比如我刚才是1，这里就是5901。

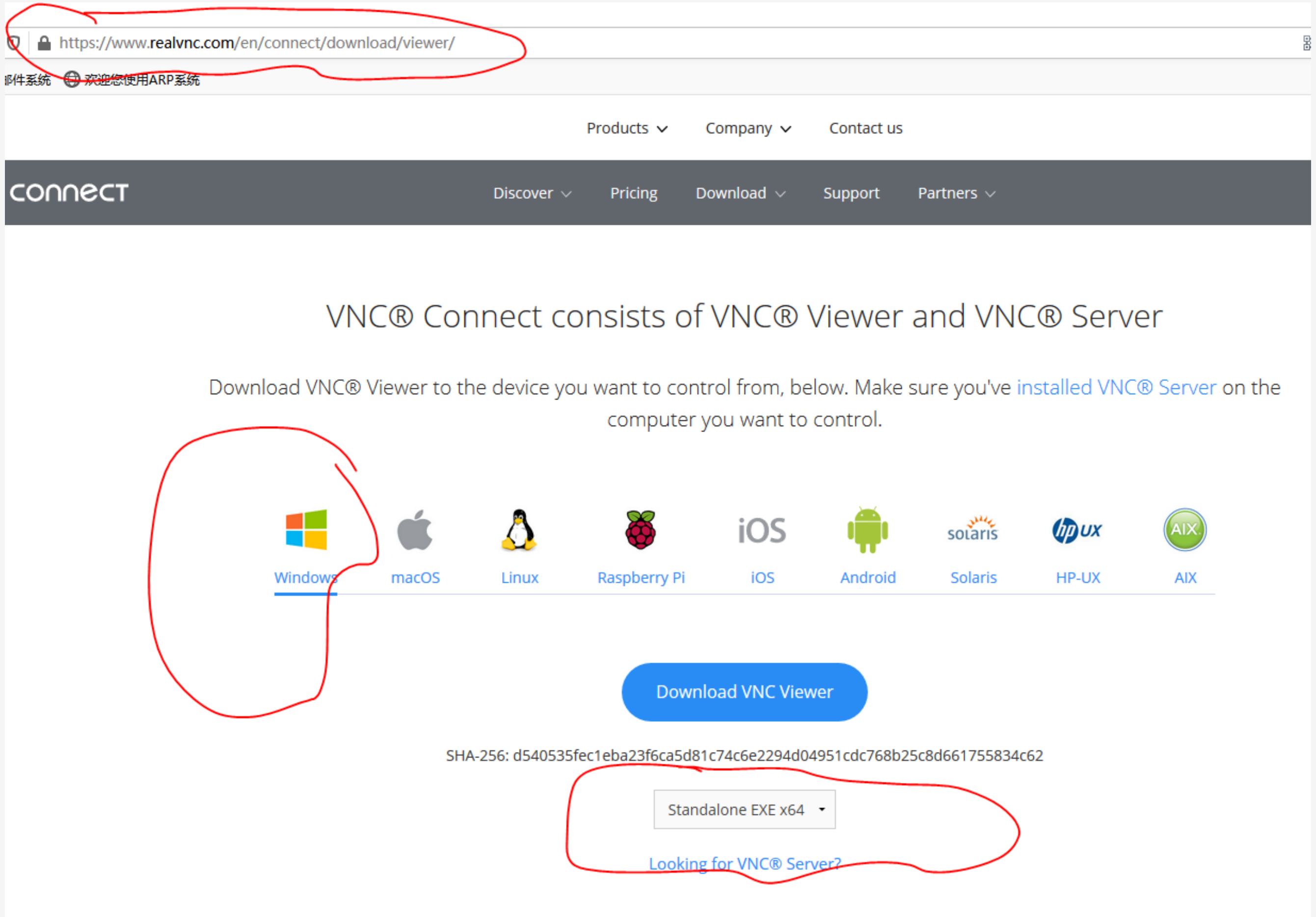
这个机器名是启动vncserver的机器



```
PS C:\Users\yuezhifeng> ssh -t -t -p 30722 yuezhifeng@hpc.cebsit.ac.cn -L 5901:localhost:5901 ssh gpu01 -L 5901:localhost:5901
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Sat May 15 16:44:51 2021 from login
[yuezhifeng@gpu01 ~]$
```

VNC远程桌面：Step 5. 下载安装客户端

下载地址：
https://www.realvnc.com/en/connect/download/viewer/



https://www.realvnc.com/en/connect/download/viewer/

VNC® Connect consists of VNC® Viewer and VNC® Server

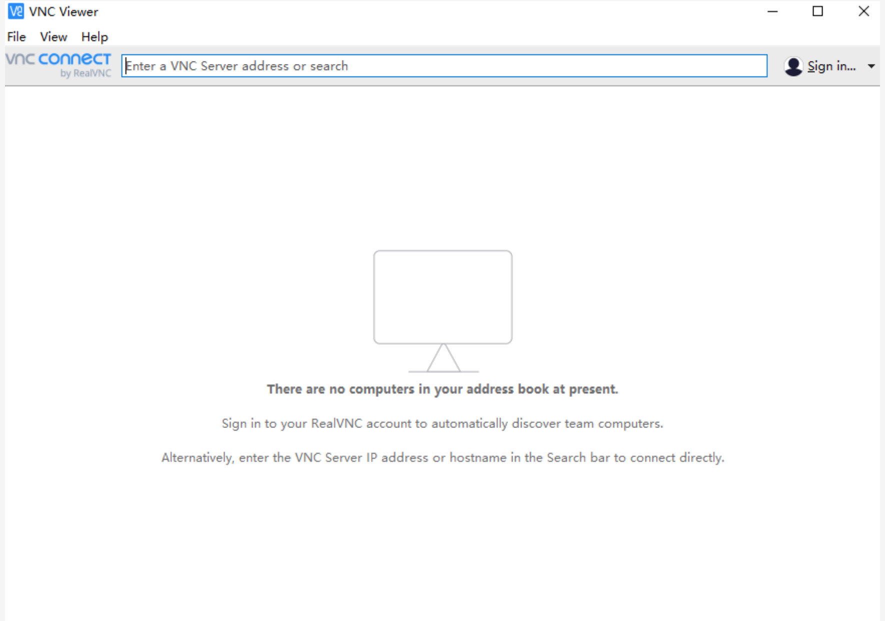
Download VNC® Viewer to the device you want to control from, below. Make sure you've installed VNC® Server on the computer you want to control.

SHA-256: d540535fec1eba23f6ca5d81c74c6e2294d04951cdc768b25c8d661755834c62

Standalone EXE x64

Looking for VNC® Server?

直接运行，打开的界面如下



Enter a VNC Server address or search

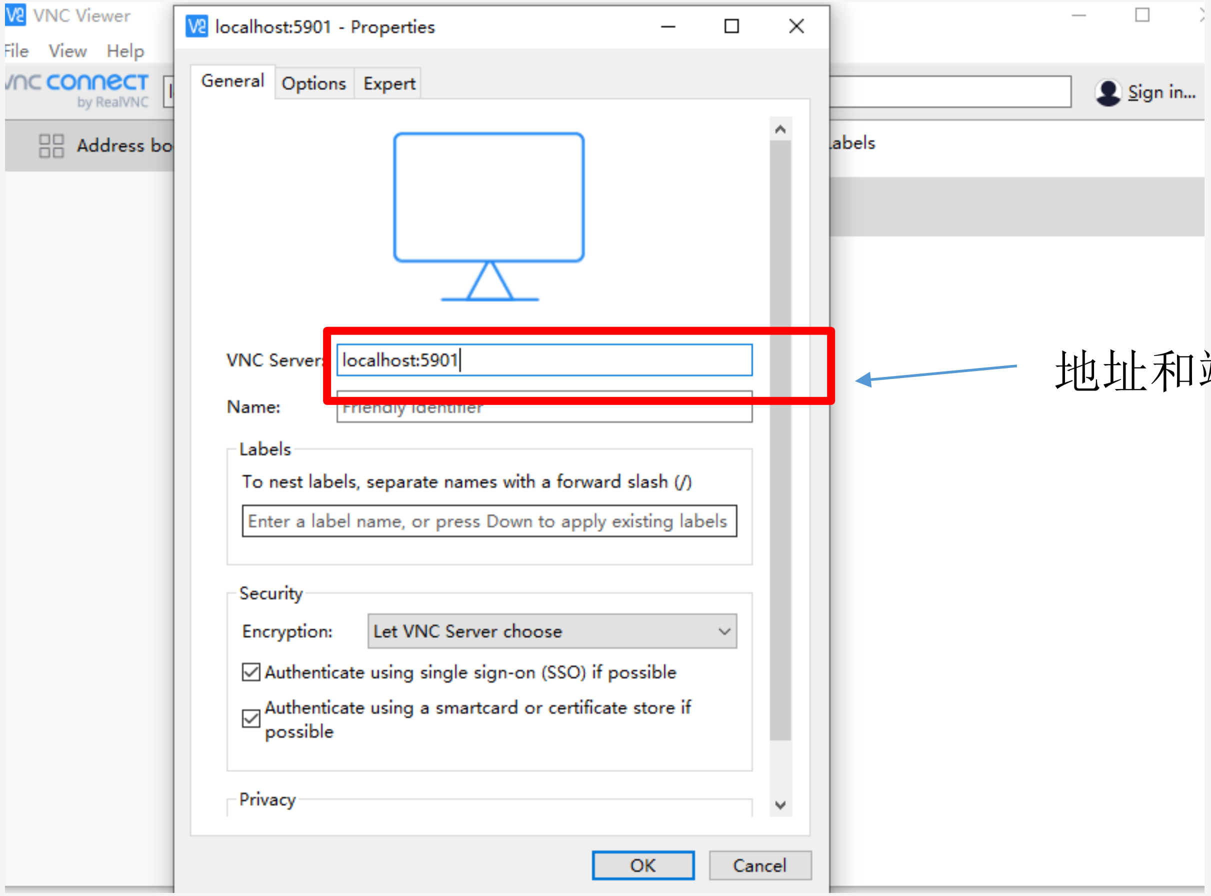
There are no computers in your address book at present.

Sign in to your RealVNC account to automatically discover team computers.

Alternatively, enter the VNC Server IP address or hostname in the Search bar to connect directly.

VNC远程桌面：Step 6a. 连接服务

通过viewer中的File菜单，新建连接



localhost:5901 - Properties

General Options Expert

VNC Server: localhost:5901

Name: Friendly identifier

Labels

To nest labels, separate names with a forward slash (/)

Enter a label name, or press Down to apply existing labels

Security

Encryption: Let VNC Server choose

☑ Authenticate using single sign-on (SSO) if possible

☑ Authenticate using a smartcard or certificate store if possible

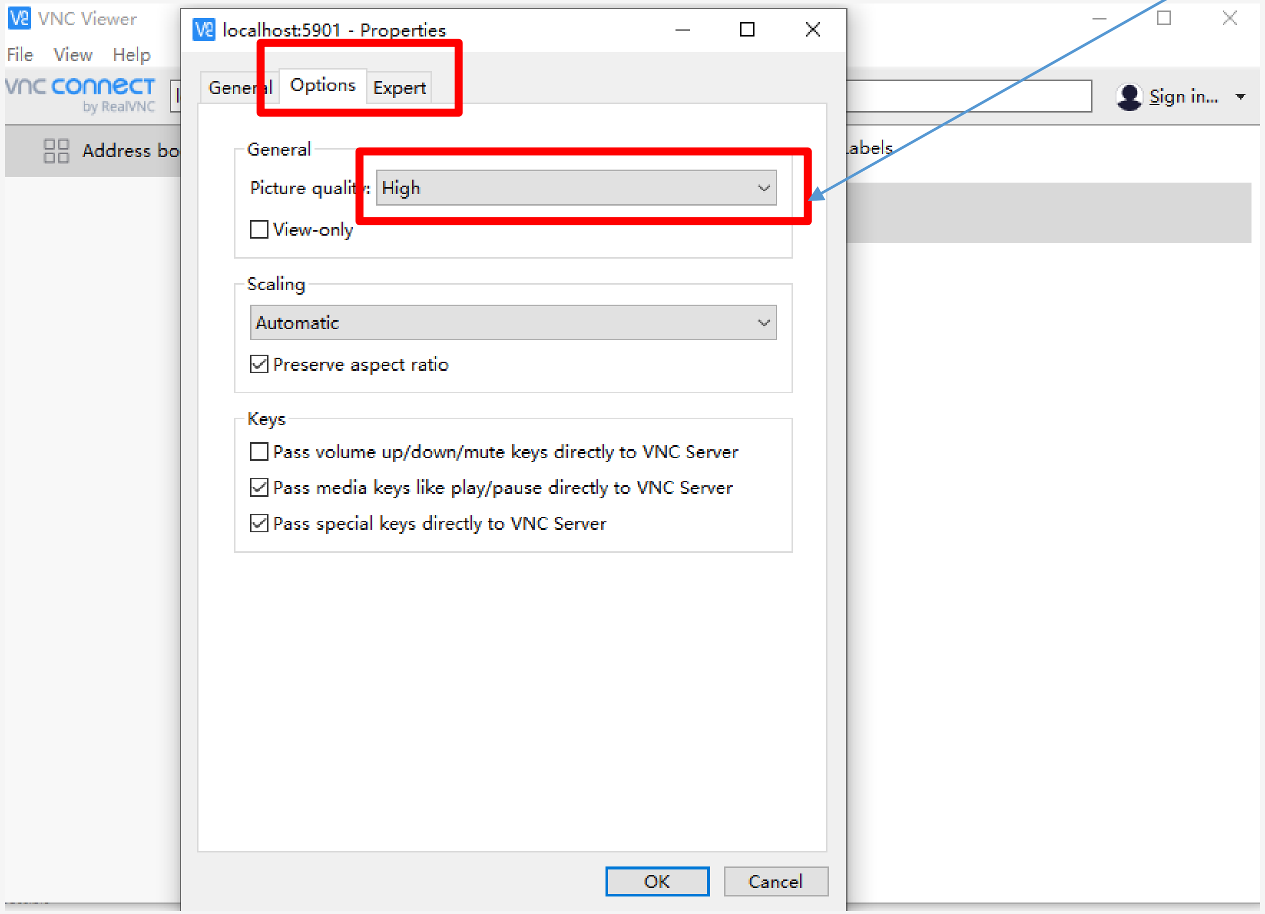
Privacy

OK Cancel

地址和端口

VNC远程桌面：Step 6b. 连接服务

通过viewer中的File菜单，新建连接



VNC Viewer

localhost:5901 - Properties

General

Picture quality: High

□ View-only

Scaling

Automatic

☑ Preserve aspect ratio

Keys

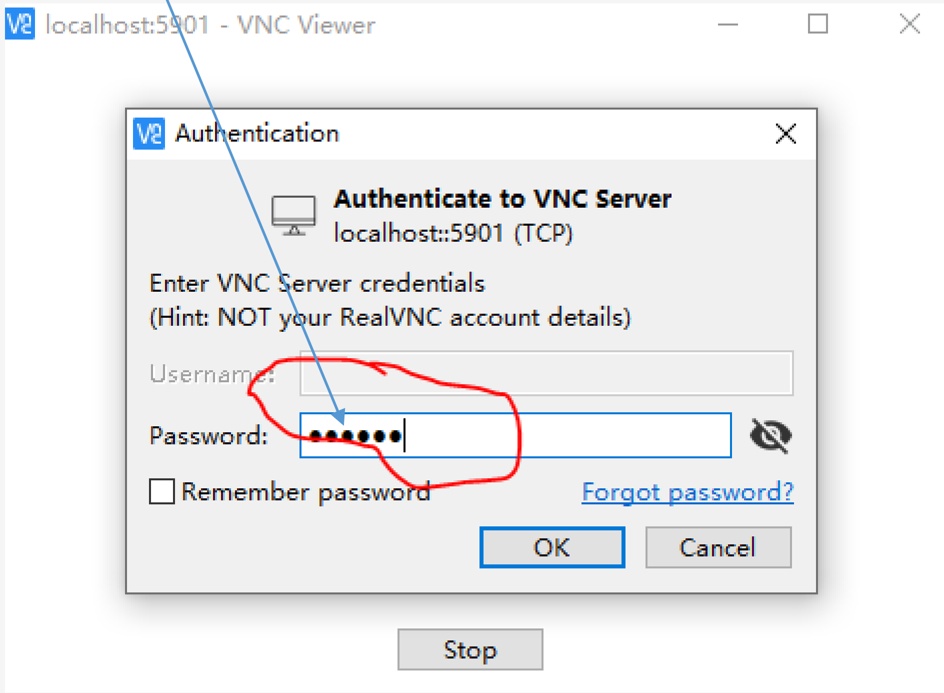
□ Pass volume up/down/mute keys directly to VNC Server

☑ Pass media keys like play/pause directly to VNC Server

☑ Pass special keys directly to VNC Server

Options里面选择High，以保证高画质

Step 3 里面设置的密码



localhost:5901 - VNC Viewer

Authentication

Authenticate to VNC Server

localhost::5901 (TCP)

Enter VNC Server credentials

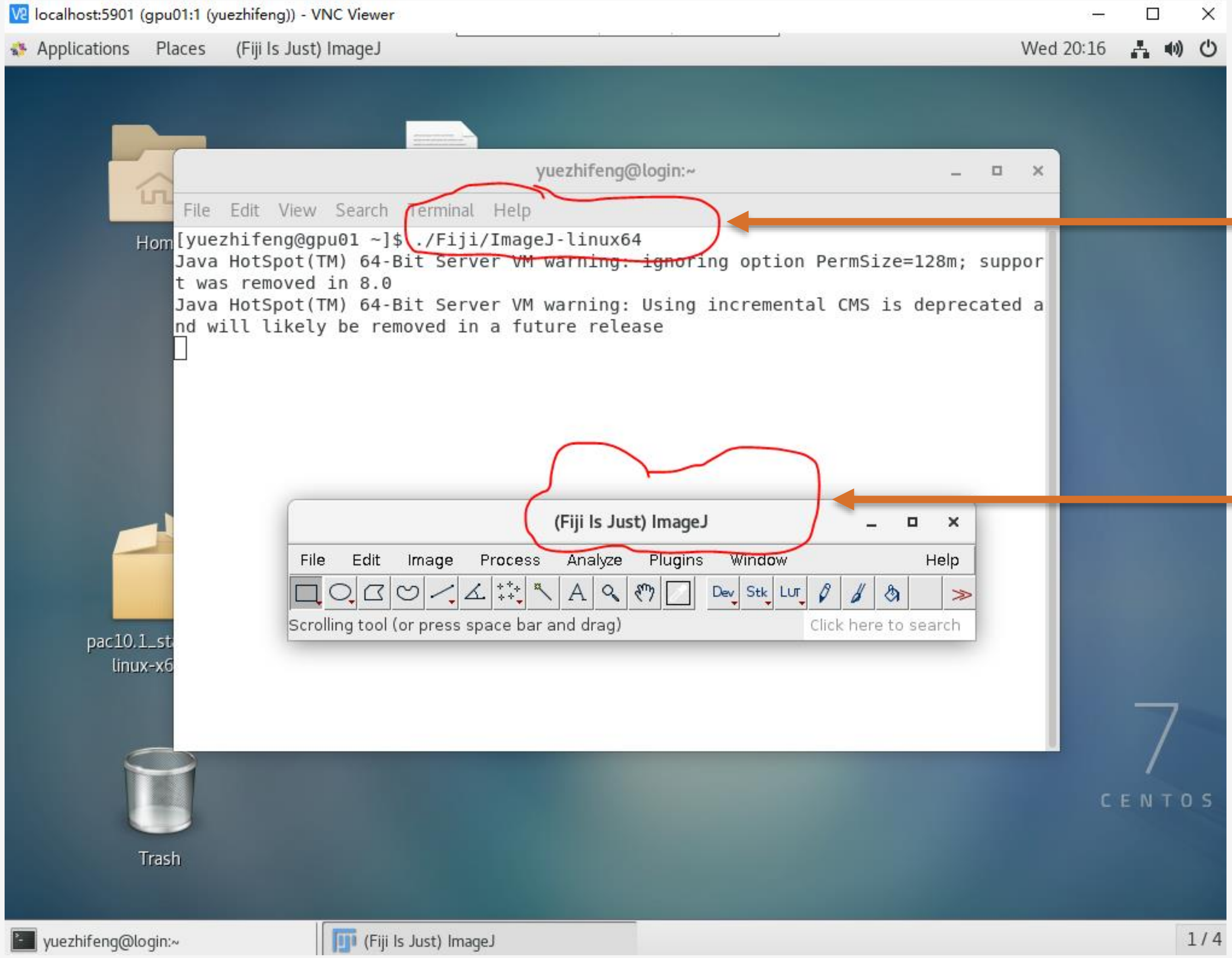
(Hint: NOT your RealVNC account details)

Username:

Password: ●●●●●●

□ Remember password

VNC远程桌面：Step 7. 成功打开GUI



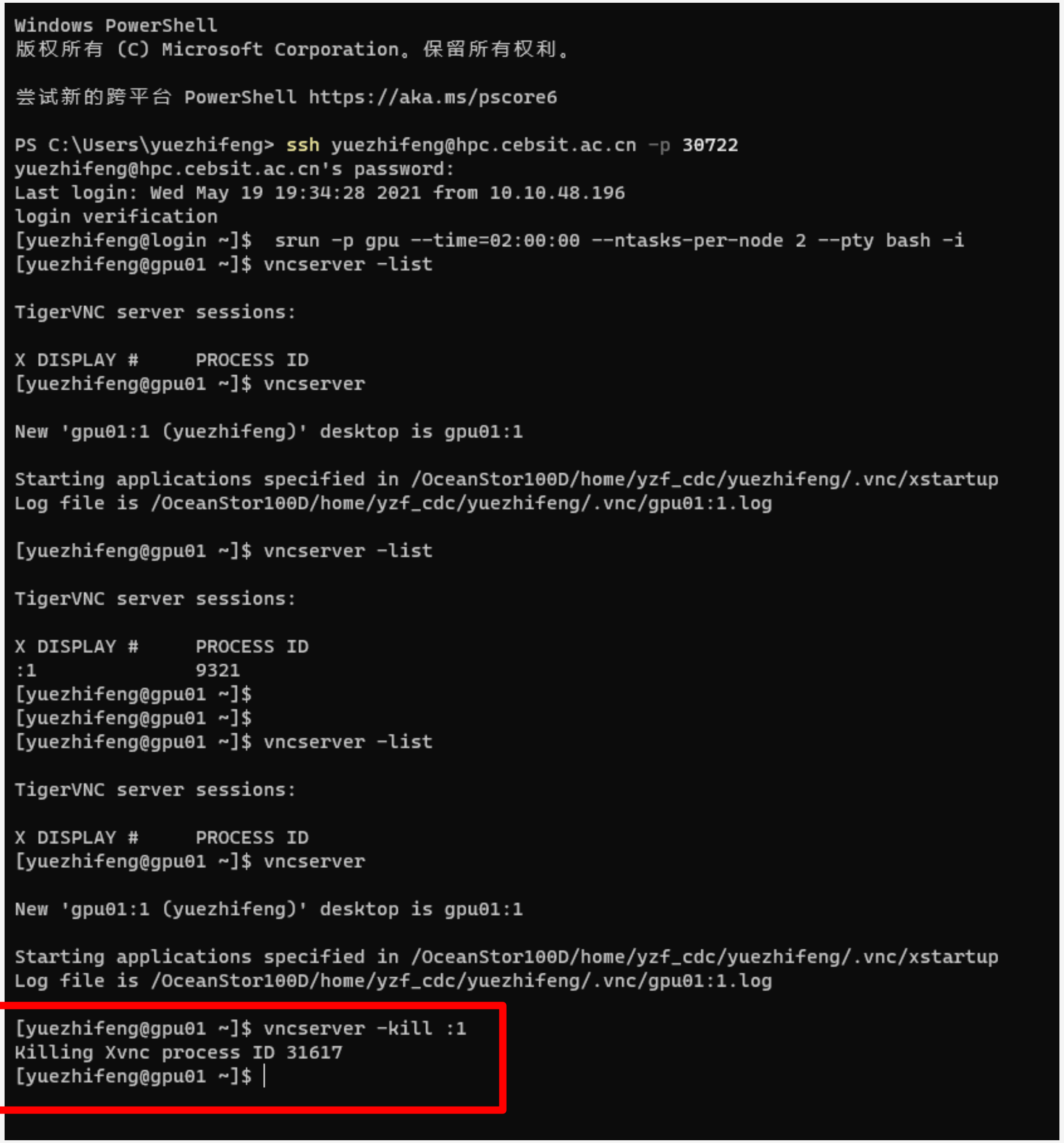
```
[yuezhifeng@gpu01 ~]$ ./Fiji/ImageJ-linux64
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option PermSize=128m; suppor
t was removed in 8.0
Java HotSpot(TM) 64-Bit Server VM warning: Using incremental CMS is deprecated a
nd will likely be removed in a future release
```

命令行启动Fiji

启动成功，可以使用大内存了，如果一开始申请了gpu卡，也许有些gpu应用也可以跑了

关闭VNC远程桌面服务

命令是：vncserver –kill :X
此例是 vncserver –kill :1



```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuezhifeng> ssh yuezhifeng@hpc.cebsit.ac.cn -p 30722
yuezhifeng@hpc.cebsit.ac.cn's password:
Last login: Wed May 19 19:34:28 2021 from 10.10.48.196
login verification
[yuezhifeng@login ~]$  srun -p gpu --time=02:00:00 --ntasks-per-node 2 --pty bash -i
[yuezhifeng@gpu01 ~]$ vncserver -list

TigerVNC server sessions:

X DISPLAY #     PROCESS ID
[yuezhifeng@gpu01 ~]$ vncserver

New 'gpu01:1 (yuezhifeng)' desktop is gpu01:1

Starting applications specified in /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/xstartup
Log file is /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/gpu01:1.log

[yuezhifeng@gpu01 ~]$ vncserver -list

TigerVNC server sessions:

X DISPLAY #     PROCESS ID
:1              9321
[yuezhifeng@gpu01 ~]$
[yuezhifeng@gpu01 ~]$
[yuezhifeng@gpu01 ~]$ vncserver -list

TigerVNC server sessions:

X DISPLAY #     PROCESS ID
[yuezhifeng@gpu01 ~]$ vncserver

New 'gpu01:1 (yuezhifeng)' desktop is gpu01:1

Starting applications specified in /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/xstartup
Log file is /OceanStor100D/home/yzf_cdc/yuezhifeng/.vnc/gpu01:1.log

[yuezhifeng@gpu01 ~]$ vncserver -kill :1
Killing Xvnc process ID 31617
[yuezhifeng@gpu01 ~]$ 
```

一 使用集群计算资源的方式

二 集群硬件和软件栈介绍

三 CPU集群容器工具介绍与容器示例

四 交互式使用计算资源的详细步骤

五 集群使用注意事项和账户申请流程

使用注意事项

千兆/万兆CEBSIT内网

不能在登陆节点做计算任务或其它高CPU消耗任务

CPU登录节点

- ✖ 计算任务
- ✖ 大文件解压缩
- ✖ 大规模编译代码
- ✖ Jupyter
- ✖ 无活动保持ssh在线
- ✖ 内网穿透

Account application

Step 1: 填写申请表，课题组/平台负责人签字

Step 2: 将申请表拍照/扫描发送至hpc@ion.ac.cn

Step 3: 集群管理员收到申请后会把集群培训资料发送给您，经过学习之后可以通过邮件或者电话预约参加考试。

Step 4: 考试通过后联系集群管理员，加入CPU集群用户微信群。

Step 5: 开通集群账号，账号开通后集群管理员会回复用户名和密码。

注：邮箱：hpc@ion.ac.cn　电话：021-64032612