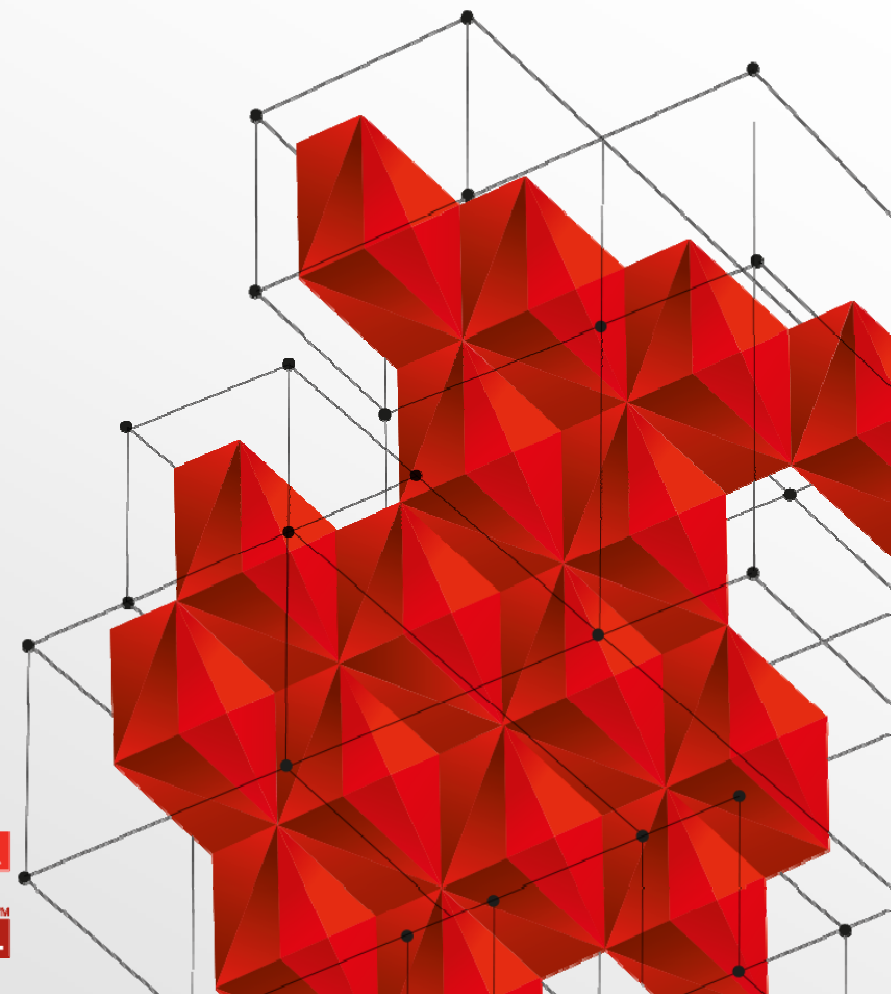
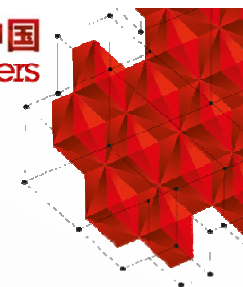


浙江大学医学院HPC集群 用户培训

培训人：
郑光奇

lenovo. FOR
THOSE
WHO DO.





目录

1、HPC集群整理环境介绍

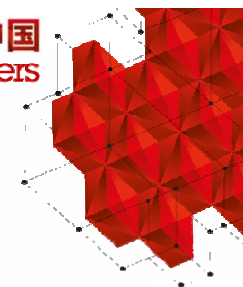
2、HPC集群登录使用方式

3、HPC集群作业调度使用方法

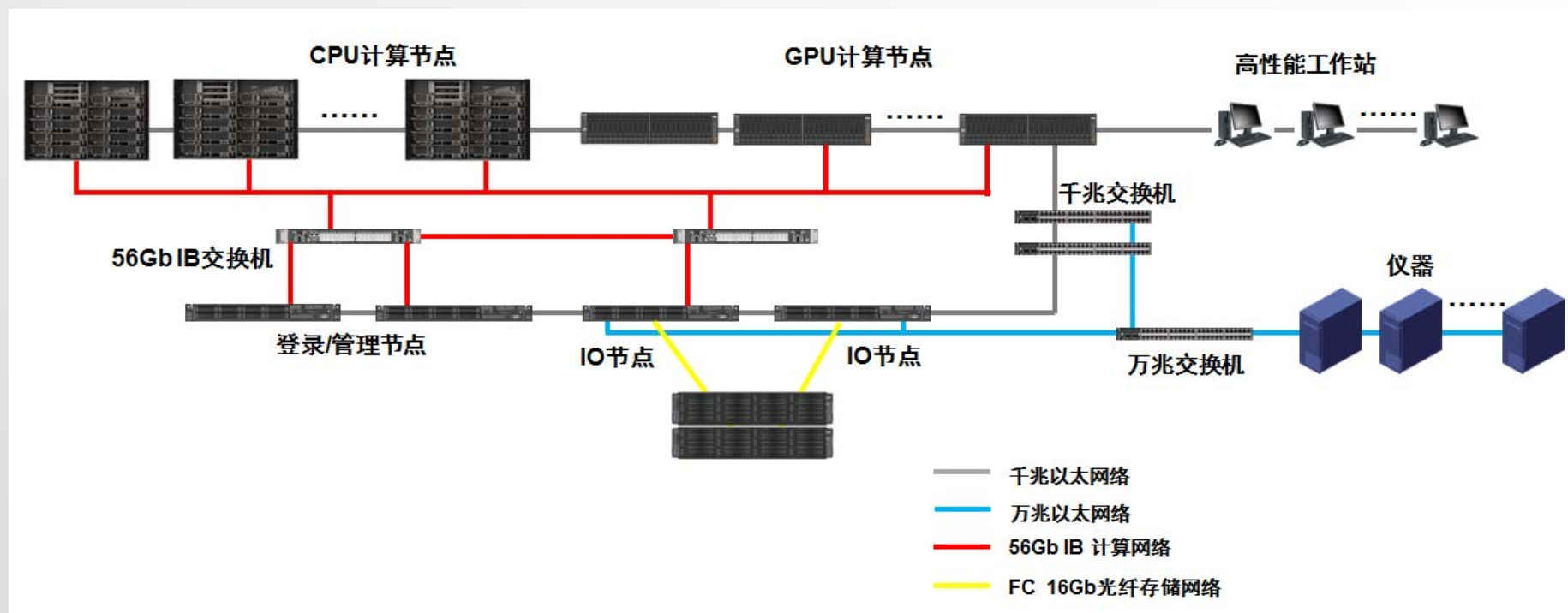
4、HPC集群软件安装

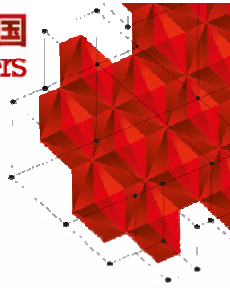
5、linux环境变量添加

6、module 加载使用环境变量



1、HPC集群整理环境介绍





1、HPC集群登录方式

- 1、校内直接登录集群ip, 校外登录需先登录vpn;
- 2、登录服务器

使用远程登陆, 使用的工具就是 ssh (linux直接登录)

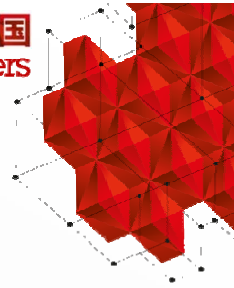
```
ssh username@ipaddress
```

这里 username 是自己的用户名, ipaddress 是服务器的 IP 地址

注: 若是需要调用图形化窗口加参数 "-X "

```
ssh -X username@ipaddress
```

使用windows 需要安装linux登录工具: xmanager、putty等



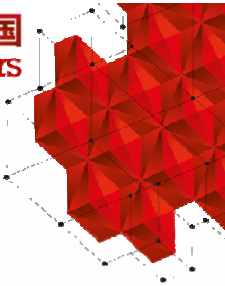
3、HPC集群软件环境介绍--intel

上传与下载

```
scp /data/path1/file user2@ip2:/data/path2/
```

```
scp user1@ip1:/data/path1/file user2@ip2:/data/path2/
```

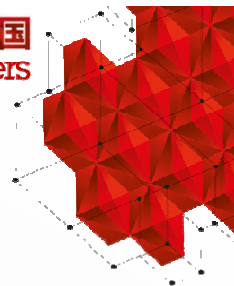
```
scp -r user1@ip1:/data/path1 user2@ip2:/data/path2/
```



3、HPC集群软件环境介绍--lsf

◆ 功能介绍

- IBM Platform LSF (Load Sharing Facility) 平台系列软件能帮助用户灵活使用应用程序，并自动操控数据流，同时LSF可以协助集群管理者监督集群的系统和资源，管理软件的使用权。
- 与以往需要安装整合多重软件包不同，IBM Platform LSF提供了统一的管理功能集，可简化应用程序集成过程，使得最终用户可以专注于开发应用程序而不是集群的管理。并且这个平台有一套综合的基于智能的、策略驱动的调度策略，方便用户使用所有运算基础设施资源，帮助保障最佳的应用性能。它具有一个统一的Web门户网站，管理员和用户均可通过其来访问和管理集群中的资源。
- 优势：
 - 集群设置简便
 - 工作负载管理
 - 工作负载和系统监控与报告
 - 支持GPU调度



3、HPC集群软件环境介绍--lsf

1、利用命令行监控作业情况

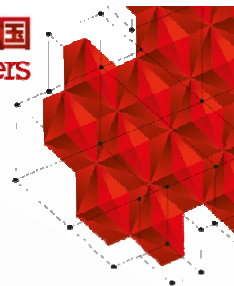
- bjobs 查看当前系统中所有状态的作业

```
[zk@log root]$ bjobs
JOBID  USER  STAT  QUEUE          FROM_HOST  EXEC_HOST  JOB_NAME  SUBMIT_TIME
19019  zk    RUN   medium_pri    log        c02n02    sleep 100  Jun 17 12:59
19020  zk    RUN   medium_pri    log        c02n02    sleep 100  Jun 17 12:59
19021  zk    RUN   medium_pri    log        c02n02    sleep 100  Jun 17 12:59
19022  zk    RUN   medium_pri    log        c02n02    sleep 100  Jun 17 12:59
```

- bjobs jobID 查看某一个jobID的作业状态

```
[zk@log root]$ bjobs 19022
JOBID  USER  STAT  QUEUE          FROM_HOST  EXEC_HOST  JOB_NAME  SUBMIT_TIME
19022  zk    RUN   medium_pri    log        c02n02    sleep 100  Jun 17 12:59
```

- bjobs -l jobID 查看该jobID作业的详细信息



3、HPC集群软件环境介绍--lsf

2、利用命令行监控作业情况

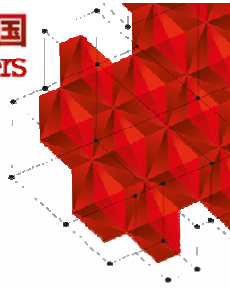
- `bhist -u all` 查看所有历史作业，默认缓存1个小时的作业情况

```
[zk@log root]$ bhist -u all
Summary of time in seconds spent in various states:
JOBID  USER  JOB_NAME  PEND  PSUSP  RUN  USUSP  SSUSP  UNKWN  TOTAL
19017  wrf    wrfrun    0      0    2094    0      0      0    2094
19018  wrf    *ar_post  1      0    1332    0      0      0    1333
```

- `bqueues` 查看集群中所有的队列信息

```
[root@mgt ~]# bqueues
QUEUE_NAME  Prio STATUS      MAX JL/U JL/P JL/H NJOBS  PEND  RUN  SUSP
priority    43  Open:Active  -   -   -   -     0     0    0    0
normal      30  Open:Active  -   -   -   -     0     0    0    0
fat-768g    30  Open:Active  -   -   -   -     0     0    0    0
fat-512g    30  Open:Active  -   -   -   -     0     0    0    0
power       30  Open:Active  -   -   -   -     0     0    0    0
tesla       30  Open:Active  -   -   -   -     0     0    0    0
geforce     30  Open:Active  -   -   -   -     0     0    0    0
quadro      30  Open:Active  -   -   -   -     0     0    0    0
interactive 30  Open:Active  -   -   -   -     0     0    0    0
idle        20  Open:Active  -   -   -   -     0     0    0    0
[root@mgt ~]#
```

- `bqueues -l fat` 查看某个队列（fat队列）的详细信息



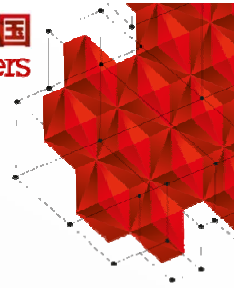
3、HPC集群软件环境介绍--lsf

3、利用命令行监控集群状态

- bhost 查看集群所有节点的状态以及可用核数

```
[root@mgt ~]# bhosts
```

HOST_NAME	STATUS	JL/U	MAX	NJOBS	RUN	SSUSP	USUSP	RSV
c01n01	ok	-	36	0	0	0	0	0
c01n02	ok	-	36	0	0	0	0	0
c01n03	ok	-	36	0	0	0	0	0
c01n04	ok	-	36	0	0	0	0	0
c01n05	ok	-	36	0	0	0	0	0
c01n06	ok	-	36	0	0	0	0	0
c01n07	ok	-	36	0	0	0	0	0
c01n08	ok	-	36	0	0	0	0	0
c01n09	ok	-	36	0	0	0	0	0
c01n10	ok	-	36	0	0	0	0	0
c01n11	ok	-	36	0	0	0	0	0
c01n12	ok	-	36	0	0	0	0	0
c02n01	ok	-	36	0	0	0	0	0
c02n02	ok	-	36	0	0	0	0	0
c02n03	ok	-	36	0	0	0	0	0
c02n04	ok	-	36	0	0	0	0	0
c02n05	ok	-	36	0	0	0	0	0
c02n06	ok	-	36	0	0	0	0	0
c02n07	ok	-	36	0	0	0	0	0
c02n08	ok	-	36	0	0	0	0	0
c02n09	ok	-	36	0	0	0	0	0
c02n10	ok	-	36	0	0	0	0	0
c02n11	ok	-	36	0	0	0	0	0
c02n12	ok	-	36	0	0	0	0	0
c03n01	ok	-	36	0	0	0	0	0
c03n02	ok	-	36	0	0	0	0	0
c03n03	ok	-	36	0	0	0	0	0
c03n04	ok	-	36	0	0	0	0	0



3、HPC集群软件环境介绍--lsf

4、利用命令行监控集群状态

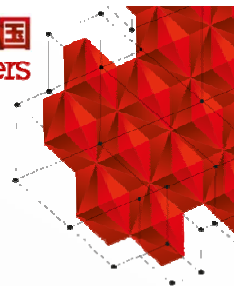
- lsid 查看集群系统LSF的具体信息

```
[root@mgt ~]# lsid
IBM Platform LSF Standard 9.1.3.0, Jul 04 2014
Copyright IBM Corp. 1992, 2014. All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

My cluster name is cluster1
My master name is login01
[root@mgt ~]# █
```

- lsload 查看所有节点的资源信息

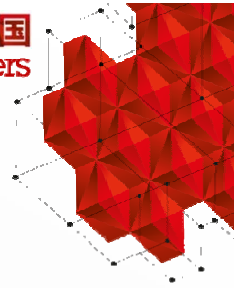
```
[root@mgt ~]# lsload
HOST_NAME      status  r15s  r1m  r15m  ut    pg  ls    it    tmp    swp    mem
gpu15          ok     0.0  0.0  0.0  0%   0.0  0    185  305G  411M  501G
login01        ok     0.0  0.1  0.1  0%   0.0  3     0    166G  62.2G  19.9G
c01n03         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
c01n06         ok     0.0  0.0  0.0  0%   0.0  0  46848  199G  62.4G  240G
c01n10         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
c02n01         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
c02n03         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
c02n06         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
c03n06         ok     0.0  0.0  0.0  0%   0.0  0  46848  198G  62.4G  240G
fat01          ok     0.0  0.0  0.0  0%   0.0  0    190  198G  62.4G  741G
```



3、HPC集群软件环境介绍--lsf

5、规范化的LSF提交脚本：

```
#!/bin/bash
#BSUB -J wrfchem1          #提交作业名
#BSUB -n 240               #指定使用的核数
#BSUB -R "span[ptile=24]"  #代表单个节点跑24核
#BSUB -o output_%J        #结果输出文件
#BSUB -e errput_%J        #错误输出文件
#BSUB -q fat              #指定提交的队列
cd /share/home/wrf/model/wrfchem/working/ter4km/zk/run2/run
mpirun -np 240 执行程序 > lsf.log
```



3、HPC集群软件环境介绍--lsf

5、利用命令行提交以及终止作业

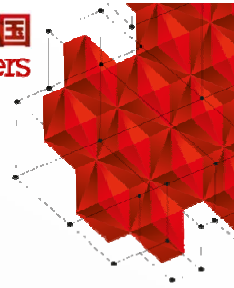
- 提交作业:

`-bsub < lsf.sh或者bsub sleep 100`

```
[zk@log root]$ bsub sleep 100
Job <19029> is submitted to default queue <medium_priority>.
[zk@log root]$ bsub sleep 100
Job <19030> is submitted to default queue <medium_priority>.
[zk@log root]$ bsub sleep 100
Job <19031> is submitted to default queue <medium_priority>.
[zk@log root]$ bsub sleep 100
Job <19032> is submitted to default queue <medium_priority>.
[zk@log root]$ bsub sleep 100
Job <19033> is submitted to default queue <medium_priority>.
[zk@log root]$ bjobs
JOBID  USER  STAT  QUEUE      FROM_HOST  EXEC_HOST  JOB_NAME  SUBMIT_TIME
19029  zk     RUN   medium_pri log        c02n01    sleep 100 Jun 17 13:27
19030  zk     RUN   medium_pri log        c02n01    sleep 100 Jun 17 13:27
19031  zk     RUN   medium_pri log        c02n01    sleep 100 Jun 17 13:27
19032  zk     RUN   medium_pri log        c02n01    sleep 100 Jun 17 13:27
19033  zk     RUN   medium_pri log        c02n01    sleep 100 Jun 17 13:27
```

删除作业:

`-bkill jobID (bkill 19033)`



3、HPC集群软件环境介绍--lsf

5、利用命令行提交GPU的作业

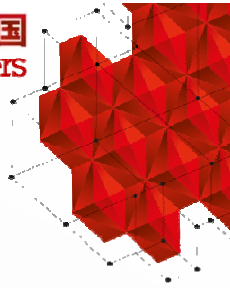
- 提交作业:

```
bsub -R "select[ngpus>0] rusage [ngpus_shared=2]" sleep 100
```

```
[lsfadmin@login01 ~]$ bsub -R "select[ngpus>0] rusage [ngpus_shared=2]" -q tesla sleep 100  
Job <525> is submitted to queue <tesla>.
```

```
[lsfadmin@login01 ~]$ bjobs  
JOBID  USER  STAT  QUEUE  FROM_HOST  EXEC_HOST  JOB_NAME  SUBMIT_TIME  
525    lsfadmi  RUN   tesla   login01    gpu13      sleep 100  Mar  6 14:09
```

选择资源中，在-R参数中使用select[]字符串来选择具有的主机GPU资源，使用rusage[]来指定lsf使用多少个GPU资源。



4、HPC集群软件安装方法

1、源码编译安装主要是用于安装一些开源的软件：

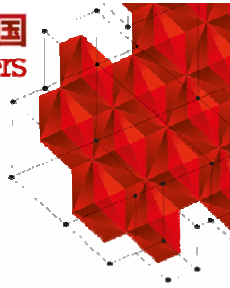
可以在软件开源社区下载到，如：<https://github.com>

源码的安装一般由3个步骤组成：

配置(configure)；

编译(make)；

安装(make install)。



4、HPC集群软件安装方法

rhel源码编译安装：

默认编译器：

```
[root@mgt ~]# which gcc
```

```
/usr/bin/gcc
```

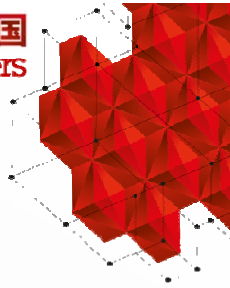
默认调用的文件位置：

/lib和/usr/lib这两个目录下的库文件

/usr/include 这么目录下的头文件

默认安装位置：/usr/local/

建议安装位置：~/software/

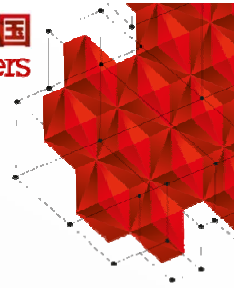


4、HPC集群软件安装方法

rhel源码编译安装的软件卸载方式：

只需要把软件安装目录删除掉就可以了；

注意点：软件安装时建议使用指定安装目录方式安装



4、HPC集群软件安装方法

2、python 模块安装方法

1、离线源码包安装

下载python软件源码包

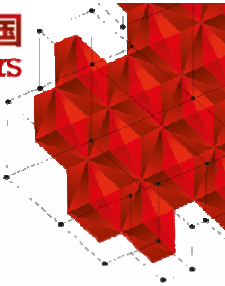
```
python setup.py install
```

2、在线安装

```
pip install 软件-版本号
```

例如在线安装Django-1.11.3:

```
pip install Django==1.11.3
```



4、HPC集群软件安装方法

修改pip的Python源站，提升下载安装的速度

修改方式：

1、临时修改方法：

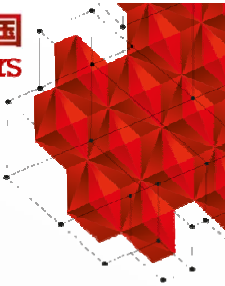
```
pip install -i https://pypi.tuna.tsinghua.edu.cn/simple gevent
```

2、永久修改方法：

修改 `~/.pip/pip.conf` (没有就创建一个)，修改 `index-url`至tuna

```
[global]
```

```
index-url = https://pypi.tuna.tsinghua.edu.cn/simple
```



4、HPC集群软件安装方法

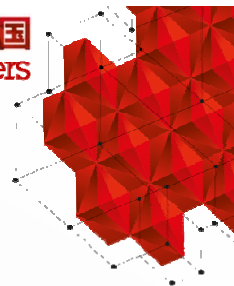
3、查看Python调用模块路径

```
#python
```

```
import os, sys
```

```
print >> sys.stderr,sys.path
```

注意点：集群上使用Python时，不要使用系统自带的Python，因为系统默认的Python目录一般不作为集群的共享目录，所以如果用系统自带的会造成，在登录节点可用，但其他节点就不可用了



5、linux环境变量设置

linux中的环境变量配置文件：

`/etc/profile`

`/etc/bashrc`

`/etc/profile.d/*.sh` & `*.csh`



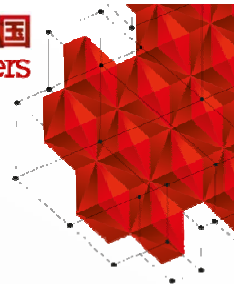
系统全局环境变量，影响所有用户

`~/.bashrc`

`~/.bash_profile`



普通用户环境变量，影响用户自己



5、linux环境变量设置

如何查看当前用户环境变量？

执行命令：`env`

`env`命令用于显示系统中已存在的环境变量，以及在定义的环境中执行指令

例：

```
[root@localhost ~]# env
```

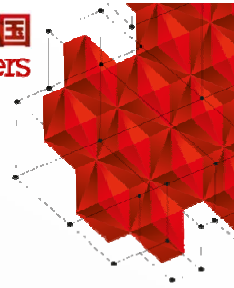
```
hostname=LinServ-1
```

```
TERM=linux
```

```
SHELL=/bin/bash
```

```
HISTSIZE=1000
```

```
.....
```

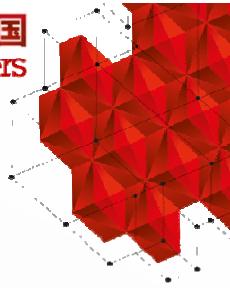


5、linux环境变量设置

如何查看当前用户环境变量？

或者：`echo $PATH`

将会把PATH这条变量的所有内容全部输出



5、linux环境变量设置

如何编写环境变量？

格式如下：

```
export PATH=$PATH:/share/apps/openmpi/bin
```

或：

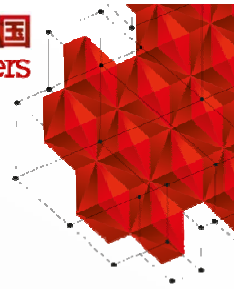
```
export PATH=/share/apps/openmpi/bin:$PATH
```

常用变量名称：

PATH

LD_LIBRARY_PATH

C_INCLUDE



6、使用module设置环境变量

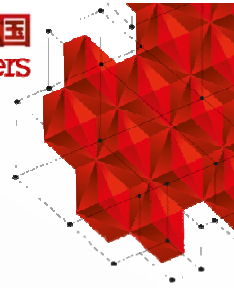
查看module下的可用模块

module av

```

----- /usr/share/Modules/modulefiles -----
dot          module-git  module-info modules      null          use.own

----- /share/Modules/modulefiles -----
anaconda2/2      gcc/4.9.3      openmpi/2.1.1
ccp4/6.5         gcc/4.9.4      pac/pac
chang/chang     gcc/5.4.0      phenix/1.9-1692
cistem/1.0.0    Gctf/0.50      python/2.6
cmake/3.9.1     Gctf-v1.06_sm_30_cu8.0/8.0 python/2.7.13
cryosparc/cryosparc  gsl/1.10      python/3.6.4
csh/csh         intel/2018     relion/2.0
cuda/7.5        make/4.2       relion/old
cuda/8.0        meteor/meteor  ResMap/1.1.4
eman1/1         modules        simple/2.5
EMAN2/2        MotionCor2/10-19-2016 spider/spider
fftw/3.3        motioncorr/v2.1 zlib/1.2.11
fltk/1.3.3     node/8.9.3
Gautomatch/0.53 openmpi/2.0.1
    
```



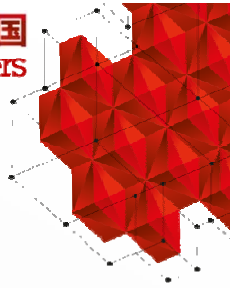
6、使用module设置环境变量

加载module下的模块

如 加载 gcc-4.9.3

module load gcc/4.9.3

```
[root@login01 ~]# module load gcc/4.9.3  
[root@login01 ~]# which gcc  
/share/apps/gcc-4.9.3/bin/gcc
```



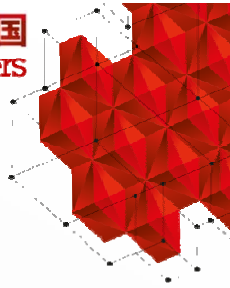
6、使用module设置环境变量

切换module模块

如 切换 gcc-4.9.3版本为gcc-4.9.4版本

module load gcc/4.9.3

```
[root@login01 ~]# module swap gcc/4.9.3 gcc/4.9.4  
[root@login01 ~]# which gcc  
/share/apps/gcc-4.9.4/bin/gcc
```



6、使用module设置环境变量

卸载module模块

如 卸载gcc-4.9.4模块

```
module unload gcc/4.9.4
```

```
[root@login01 ~]# module unload gcc/4.9.4  
[root@login01 ~]# which gcc  
/usr/bin/gcc
```

Thank You

