

User Guide for using LHAASO Computing Cluster at IHEP

Haibo Li(李海波)

IHEP-CC

lihaibo@ihep.ac.cn



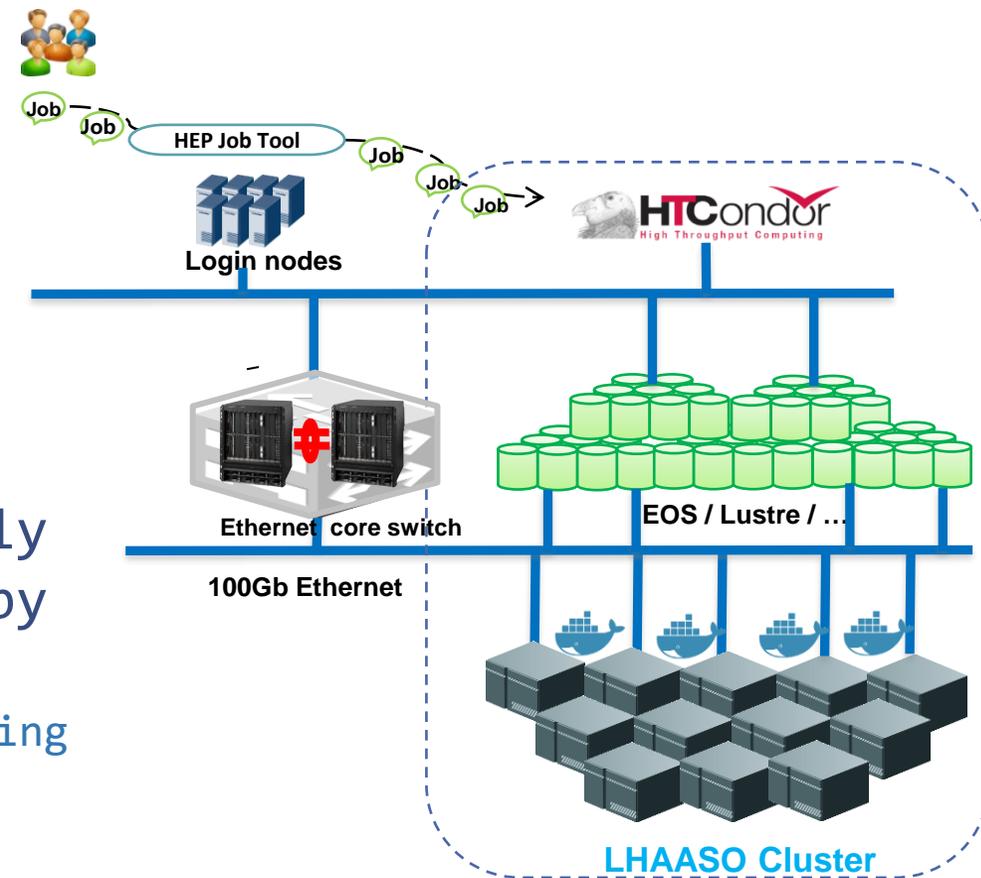
Outline

- Overview of LHAASO Cluster at IHEP
- Accounts
- Run Jobs
- Store Data
- FAQ
- How To Ask for Help



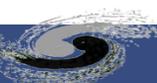
Overview of LHAAO Cluster at IHEP

- User: 446
- CPU: 12,182 CPU cores
- Storage: ~48 PB
 - EOS: 47.5PB, 40.8PB used
 - Home: 79TB, 53.7TB used
- Running HTCondor Cluster
 - User job priority - adjusted dynamically
 - based on the sum of CPU resource usage by users in the recent time period
 - Priority is calculated by exponentially decreasing the cumulative job run time over a specified period (3 days)
 - The higher the CPU usage by a user, the lower their job priority



Important changes during summer maintenance at IHEP

- All computing cluster updates from Centos 7.9 to **Alma 9.4**
- Your program should be recompiled
- Login node uses **lxlogin.ihep.ac.cn** instead of `lxslc7.ihep.ac.cn`
- Lxslc7 will be unavailable by **12th August**, please migrate your crontab jobs as soon as possible



Apply for a cluster account

- Step 1: Apply for an IHEP Single Sign On (SSO) Account from <https://login.ihep.ac.cn/>
- Step 2: Apply for IHEP Cluster account by your IHEP SSO account
 - Access: <https://login.ihep.ac.cn/afsApply.jsp>
- Step 3: Waiting for the notice email which would inform your account is ready
 - Your apply needs to be approved by the computing coordinator (Chaoyong Wu, wucy@ihep.ac.cn)
 - Your account is ready within 30 mins after the approval
 - Once a cluster account is created, the corresponding directories /afs, /workfs2, eos/user, and /home/lhaaso are created by default

IHEP unified authentication Home Forgot password Help

Registration

* Email

* Name

* Gender Male Female

* Identity IHEP Users Yes No

* Password
Weak Good Strong

* Confirm Password

Country or Region

* Home Institute

* Security Control Collaboration Contact Person

Collaboration (optional) Status

* Image Text Refresh

Apply for a cluster account

- Step 1: Apply for an IHEP Single Sign On (SSO) Account from <https://login.ihep.ac.cn/>
- Step 2: Apply for IHEP Cluster account by your IHEP SSO account
 - Access: <https://login.ihep.ac.cn/afsApply.jsp>
- Step 3: Waiting for the notice email which would inform your account is ready
 - Your apply needs to be approved by the computing coordinator (Chaoyong Wu, wucy@ihep.ac.cn)
 - Your account is ready within 30 mins after the approval
 - Once a cluster account is created, the corresponding directories /afs, /workfs2, eos/user, and /home/lhaaso are created by default

IHEP unified authentication Home Forgot password Help

Registration

* Email

王 更改

统一认证账号 simpler1996@163.com (已验证)

用户名: simpler1996

密码: ***** 更改密码

账号安全

密保邮箱 (已设置)
设置并验证密保邮箱后, 您可以使用密保邮箱找回密码。
simpler1996@163.com 更改

VPN

VPN服务
申请VPN, 您可以使用VPN账号远程办公。高能所人员申请VPN 非高能所人员申请VPN

AFS 申请集群账号
申请集群账号 申请

* Security Control

Collaboration (optional)

Status

Tip 1: You must have SSO account first

Tip 2: Cluster user password is SAME as that of SSO account

Apply for a cluster account

- Step 1: Apply for an IHEP Single Sign On (SSO) Account from <https://login.ihep.ac.cn/>
- Step 2: Apply for IHEP Cluster account by your IHEP SSO account
 - Access: <https://login.ihep.ac.cn/afsApply.jsp>
- Step 3: Waiting for the notice email which would inform your account is ready
 - Your apply needs to be approved by the computing coordinator (Chaoyong Wu, wucy@ihep.ac.cn)
 - Your account is ready within 30 mins after the approval
 - Once a cluster account is created, the corresponding directories /afs, /workfs2, eos/user, and /home/lhaaso are created by default

IHEP unified authentication Home Forgot password Help

Registration

* Email

* Name

* Gender Male Female

* Identity IHEP Users Yes No

* Password
Weak Good Strong

* Confirm Password

Country or Region

* Home Institute

* Security Control Collaboration Contact Person

Collaboration (optional) Status

* Image Text Refresh

The Process to Run a Job at LHAASO Cluster



The Process to Run a Job at LHAASO Cluster



- 4 nodes are named as `lxlogin.ihep.ac.cn` with load balance policy
 - `lxlogin001`, `lxlogin002`, `lxlogin003`, `lxlogin004`
 - Login to `lxlogin` via ssh tool: `xshell`, `mobaxterm`, `windows terminal`, `tabby`
 - `ssh lxlogin.ihep.ac.cn`
 - Reach your home directory after the success login
 - The operating system is Alma 9.4

Tip 1: run "hostname" to get login node name

Tip 2: If you want to add second linux group, access <http://ccsinfo.ihep.ac.cn>



- Code and debug your program at lxlogin
 - Code and compile your program
 - Run your program at lxlogin.ihep.ac.cn directly
 - Make sure your program runs well
 - The result is right
 - The memory is suitable (default mem is 2-3.5GB)
 - Estimate the program running time
- Code your job script
 - Recommend to code bash job script
 - Set executable permission for the job script

```
chmod +x job.sh
```





- Hep Job tool provided by IHEP-CC: `hepjob`
 - Simplified the user job management

- Add it to your `$PATH`
 - Add it to `~/.bashrc`

```
export PATH=/cvmfs/common.ihep.ac.cn/software/hepjob/bin
```

- Submit job
 - `hep_sub job.sh`
 - Default job queue is the main group that your account belong to
 - Those who is not belong to “lhaaso”, needs to add parameter `-g lhaaso`
 - `hep_sub -g lhaaso job.sh`
 - To check the group you are belong to, run “`id`”
 - Some useful parameters:
 - `hep_sub -mem 4096 -wt long job.sh` # big memory or long running time
 - `hep_sub -os CentOS7 job.sh` # Run Centos7 jobs in alma computing nodes

```
$ hep_clus -g lhaaso --walltime
```

```
Walltime Limit INFO for group 'lhaaso':  
'default' job walltime limit is: 15.0 hours.  
'long' job walltime limit is: 720.0 hours.  
'mid' job walltime limit is: 100.0 hours.  
'short' job walltime limit is: 30.0 minutes.  
'test' job walltime limit is: 5.0 minutes.
```

Tip 1: setup your environment before submission. The env parameters will be transferred to the work node the job will run.



- Query your job status

- `hep_q -u userid`
- `hep_q -u userid -run # query the jobs in running status`

- If job is held

- `hep_q -u userid -hold #query the holding reason`

- Example: `14712.0 username 08/01 18:40 H Error from slot49@lhws168.ihep.ac.cn: Job has gone over memory limit of 4101 megabytes. Peak usage: 4557 megabytes.`

- `hep_release jobid # release your holding job to the job queue`

Tip 1: finished job id is not be showed by hep_q



- Check job logs:
 - output and error log are saved at the submission directory by default
- Check program result: new data file etc.

Tip 1: No logs given with the submission like

```
hep_sub -o /dev/null -e /dev/null job.sh
```

HTCondor job batch submission

- Submit multiple jobs at once, with fast submission speed
- Submission Example:
 - Assume there is a batch of job scripts: job_0.sh, job_1.sh, ..., job_99.sh
 - To submit these jobs, you can use the following command:
 - `$ hep_sub -n 100 job_"{ProcId}".sh`
 - If the job numbering does not start from 0, but from n, then `"{ProcId}"` should be changed to `"{ProcId+n}"`.
 - If you prefer the job numbers to be written in multiple digits, such as 0001, 0002, ..., 9999, you can also use batch submission:
 - `"{ProcId}" -> "{ProcId+1:%04d}"`



LHAASO Storage at IHEP Cluster

- All storage of LHAASO are distributed file systems: server - client
- All the directories and files are exactly the same from the view of lxlogin and worker nodes

Aim	Usage	Mount point	Access way	Capacity/Files Quota	Backup
Exp. Software	LHAASO software	/cvmfs/lhaaso/	Access directly	Read only for normal user	Yes
Exp. Data	Store exp. data	/eos/lhaaso	Access via xrootd		No
User Home	User home directory	/home/lhaaso/username	Access directly	200GB/500,000	recycle
User Data	Store user data	/eos/user/[a-z]/username	Access via xrootd	1TB / 250,000	recycle
Other	Space provided by IHEP-CC	1. /afs/users/[a-z]/username 2. /workfs2/lhaaso/username	Access directly	1. 500MB/none 2. 5GB/50,000	1. Yes 2. Yes

EOS Storage – XRootD Usage

- XRootD: user needs to set EOS instance

- Access files by the EOS commands

```
export EOS_MGM_URL=root://eos01.ihep.ac.cn/
```

Or

```
eos root://eos01.ihep.ac.cn/ ls /eos/lhaaso
```

- Access files by XRootD commands

```
xrdfs root://eos01.ihep.ac.cn ls /eos/lhaaso
```

- Recycle: support query, clean, recover

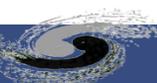
- \$ eos recycle [ls / purge / restore]

- Files in recycle are kept only **3 days**

- EOS usage: [EOS manual](#)

EOS Command and Linux Command Comparison Table

Eos comm. (recommend using)	Linux comm. (unavailable to EOS)	Description
eos ls	ls	List file name
eos cp	cp	Copy file
eos mv eos file rename	mv	Rename file
eos cp /eos/user/myfile - cat	cat	View file content
eos cp /eos/user/myfile - tail	tail	View file content
eos mkdir	mkdir	Create directory
eos touch	touch	Create file
eos newfind -f /eos/mypath	none	Search for a list of all files in a specific directory (including subdirectories)
eos newfind -d /eos/mypath	none	Search for a list of all directories in a specific directory (including subdirectories)
eos ln	ln	Create file softlink
eos quota	none	Check personal storage space usage (/eos/user/a-z/username)
eos stat	-f	check if a file exists and return an empty value if it does not exist
eos file info	stat	To view a file's modification time (modify) and change status time (change),



How to Access EOS File in Your Program

- Access from ROOT
 - Supports XRootD access. Three ways to create TFile objects
 - declare: `Tfile(PATHNAME) -- unsupport XRootD`
 - New file: `new Tfile(PATHNAME) -- unsupport XRootD`
 - **Open: `Tfile :: Open(PATHNAME) -- support XRootD`**
 - Example:
 - directly open ROOT format file by using `TFile::Open`

```
TFile *filein = TFile::Open("root://eos01.ihep.ac.cn//eos_absolute_path_filein_name.root")
```

- Read or write non-ROOT-format files by using the TFile class and have to append an additional parameter to the file name
 - `?filetype=raw`
 - optimize the access by preloading a data block size worth of data into memory to accelerate file access speed
- Access EOS files from python

```
f=ROOT.TFile.Open("root://eos01.ihep.ac.cn//eos_absolute_path_fileout_name.root")
```

```
void rawfile(){  
  
    int size;  
  
    char buf[1024];  
  
    TFile *rf = TFile::Open("root://eos01.ihep.ac.cn//eos/user/c/chyd/set.log?filetype=raw");  
  
    size = rf->GetSize();  
  
    printf("size is %d\n", size);  
  
    memset(buf, 0, 1024);  
  
    rf->ReadBuffer(buf, 1024);  
  
    printf("%s\n", buf);  
  
    rf->Close();  
  
}
```

hadd with xrootd support

- `$ source /cvmfs/lhaaso.ihep.ac.cn/anysw/slc5_ia64_gcc73/external/envf.sh`
- `$ histadd -h`

```
$ histadd -h
Function: An alternative "hadd" made by <zhiguo.yao@ihep.ac.cn>
SYNOPSIS:
  histadd [ -o outfile ]
  [ -topdir | +topdir ] [ -folder folder1 [ -folder folder2 ... ] ]
  [ -tree | +tree ]
  [ -list listfile1 [ -list listfile2 ... ] ]
  [ -inclhist histname1 [ -inclhist histname2 ... ] ]
  [ -exclhist histname1 [ -exclhist histname2 ... ] ]
  [ filename1 [ filename2 ... ] ]

BRIEF EXPLANATIONS:
-topdir: Add objects in the top directory of every root file (default).
+topdir: Ignore all objects in the top directory.
-folder folder: Activate adding the TFolder "folder".
*** Note: Only folders specified by option "-folder folder" are added.
-tree: Add TTree objects (default).
+tree: Ignore all TTree objects.
-inclhist histname: include histname in the top directory. Default: all.
-exclhist histname: exclude histname in the top directory. Default: nothing.
-list listfile: Add every lines in "listfile" to the input list.
*** Note: Lines with leading "#" will be ignored!
filename: Add "filename" to the input list.
*** Note: (Many) list files and file names can be specified simultaneously.
*** Note: EOS files will be automatically accessed via the xrootd protocol.
```

- Histadd usage: <http://afsapply.ihep.ac.cn/cchelp/zh/experiments/LHAASO/>



Suggestions to EOS Usage

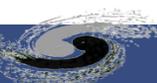
- Do not store too many files or scripts (tens of thousands or more) in a single directory
- Instead, create subdirectories following a specific pattern and place the files within these subdirectories
- Limit the number of files in a single directory to no more than 3000
- Avoid using wildcard operations
 - Avoid using operations like `eos ls *` or `eos rm *` in your tasks. If you need to access the `/eos` directory, use the absolute path with `eos ls` for faster performance
 - Avoid using operations like `'hadd *.root'` when there are a large number of `'*.root'` files, use `histadd` instead of `hadd`
- Access files on eos via XRootD



EOS Directories of LHAASO Data

- Each directory contains ‘km2a,’ ‘wcda,’ ‘wcdapls,’ ‘wfcta,’ and each directory is managed by the respective experiment specialist. Data is organized and stored in directories based on dedicate directory hierarchy.

Directory	Usage	Comment
/eos/lhaaso/raw	Store raw data	Responsible by IHEP-CC
/eos/lhaaso/spt	Store single particle data	Responsible by IHEP-CC
/eos/lhaaso/decode	Store decode data	Responsible by LHAASO
/eos/lhaaso/rec	Store reconstruction data	Responsible by LHAASO
/eos/lhaaso/cal	Store calibration parameter data	Responsible by LHAASO
/eos/lhaaso/monitor	Store data quality monitoring data, as well as moon shadow and Crab monitoring data	Responsible by LHAASO
/eos/lhaaso/simulation	Store simulation data	Responsible by LHAASO



Module

- Add modulefile path
 - \$ module use /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles
- Find available software and versions
 - \$ module avail

```
[l1haibo@ixlogin001 ~]$ module avail
----- /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles -----
anaconda/24.3.0      elegant/1.0          intel_oneapi/compiler-rt/latest      intel_oneapi/fort/latest             intel_oneapi/tbb/latest              python/3.10.14       ucx/1.17.0-gcc11
cmake/3.18.4         epics/7.0.7          intel_oneapi/compiler-rt32/2024.1.0  intel_oneapi/fort32/2024.1.0         intel_oneapi/tbb32/2021.12          python/3.11.8        vmd/1.9.4
cmake/3.26.4         fftw/3.3.10-gcc11    intel_oneapi/compiler-rt32/latest    intel_oneapi/fort32/latest           intel_oneapi/tbb32/latest           python/3.12.2        warpx/24.02
cmake/3.29.1         gcc/7.5.0            intel_oneapi/compiler/2024.1.0       intel_oneapi/intel_ipp_ia32/2021.11  intel_oneapi/vtune/2024.1           pytorch/2.0.1-cu117-py310
cp2k/2023.1-gcc11   gcc/9.5.0            intel_oneapi/compiler/latest          intel_oneapi/intel_ipp_ia32/latest    intel_oneapi/vtune/latest           pytorch/2.2.0-cu121-py310
cp2k/2024.1-gcc11   gcc/10.4.0           intel_oneapi/compiler32/2024.1.0     intel_oneapi/intel_ipp_intel64/2021.11  lammps/2024.02.07                  quickpic/develop
cuda/10.1-cvmfs     gcc/11.4.0           intel_oneapi/compiler32/latest        intel_oneapi/intel_ipp_intel64/latest  lapack/3.11.0-gcc11                 scalapack/2.2.1
cuda/11.0-cvmfs     gcc/12.3.0           intel_oneapi/dal/2024.0.0             intel_oneapi/intel_ippc_ia32/2021.11  lume-astro/0.6.1                    scow/1.0
cuda/11.1-cvmfs     glmic/2.0             intel_oneapi/dal/latest               intel_oneapi/intel_ippc_ia32/latest    molpro/2015-gcc11                   tensorflow/2.4.0-cu110-py38
cuda/11.2-cvmfs     gromacs/2023.4-gcc11 intel_oneapi/debugger/2024.1.0        intel_oneapi/intel_ippc_intel64/2021.11  mpi/mpich/4.1.3                      tensorflow/2.11.0-cu112-py310
cuda/11.7-cvmfs     hdf5/1.10.11         intel_oneapi/debugger/latest          intel_oneapi/intel_ippc_intel64/latest  mpi/mpich/4.2.1                      tensorflow/2.15.0-cu122-py310
cuda/12.2-cvmfs     intel_oneapi/advisor/2024.1  intel_oneapi/dev-utilities/latest     intel_oneapi/mkl/2024.1               mpi/mvapich/2.3.7                    tensorflow/2.16.1-cu123-py310
cuda/12.4           intel_oneapi/advisor/latest  intel_oneapi/dev-utilities/latest     intel_oneapi/mkl/latest               mpi/mvapich/3.4.3                    TensorRT/7.2.3.4-cuda11.0-cudnn8.1
cuda/12.4-cvmfs     intel_oneapi/ccl/2021.12.0  intel_oneapi/dnnl/2.4.0               intel_oneapi/mkl32/2024.1             openmpi/4.1.4-gcc11                  TensorRT/7.2.3.4-cuda11.1-cudnn8.1
cuDNN/8.0.5-cuda11.0  intel_oneapi/ccl/latest    intel_oneapi/dnnl/latest              intel_oneapi/mkl32/latest             openmpi/5.0.1-cuda12.4-gcc11         TensorRT/8.6.1.6-cuda11.8
cuDNN/8.1.1-cuda11  intel_oneapi/compiler-intel-llvm/2024.1.0  intel_oneapi/dpct/2024.1.0           intel_oneapi/mpi/2021.12              orca/5.0.4-gcc11                      TensorRT/8.6.1.6-cuda12.0
cuDNN/8.9.7-cuda11  intel_oneapi/compiler-intel-llvm/latest    intel_oneapi/dpct/latest              intel_oneapi/mpi/latest               python/2.7.18                          TensorRT/10.0.1.6-cuda11.8
cuDNN/8.9.7-cuda12  intel_oneapi/compiler-intel-llvm32/2024.1.0  intel_oneapi/dpl/2022.5               intel_oneapi/oclfpga/2024.1.0         python/3.7.16                          TensorRT/10.0.1.6-cuda12.4
cuDNN/9.0.0-cuda11  intel_oneapi/compiler-intel-llvm32/latest    intel_oneapi/dpl/latest               intel_oneapi/oclfpga/latest           python/3.8.19                          tfpwa/tf2.4.0
cuDNN/9.0.0-cuda12  intel_oneapi/compiler-rt/2024.1.0           intel_oneapi/fort/2024.1.0           intel_oneapi/tbb/2021.12              python/3.9.18                          ucx/1.13.1-gcc11
```

- Load software
 - \$ module load <module_file>
- Check loaded software
 - \$ module list
- Unload software
 - \$ module unload <module_file>
- Module usage: <http://afsapply.ihep.ac.cn/cchelp/zh/others/module/>

Container

- Hep_container: a container client tool developed based on the Apptainer container, suitable for IHEP computing cluster
 - meeting the needs of users for various operating system versions and environments
- Add hep_container path
 - `$ export PATH=/cvmfs/container.ihep.ac.cn/bin/:$PATH`
- hep_container usages
 - images、shell、exec
- Start a centos79 container:
 - `$ hep_container shell CentOS79`

```
[lihaibo@lxlogin001 ~]$ hep_container shell CentOS79
Singularity> cat /etc/redhat-release
CentOS Linux release 7.9.2009 (Core)
Singularity> █
```

- Container usage: <http://afsapply.ihep.ac.cn/cchelp/zh/local-cluster/container/>

FAQ –Job Issues

- Job has been waiting in queue for a long wait time
 - The HTCondor adjusts the priority of users who have recently run a large number of jobs in real-time to ensure fairness among users
 - Peak job times – many users and many jobs
 - when high-priority public service jobs are running, resources become extremely tight, leading to extended wait time
 - long-duration or high-memory jobs, may also have longer queue times due to limited available resources
- My job is being holding
 - Run “`hep_q -u userid -hold`” to check the reasons for it being held
 - The most possible reason:
 - attempting to write job data or logs to directories like AFS or /workfs2 -- read-only on worker nodes
 - job exceeds memory limits
 - Modify your program to reduce memory
 - resubmit it as a high-memory job: `hep_sub -mem 4096 job.sh`

FAQ - Storage Issues

- Fail to write my data directory
 - Possible reasons
 - Exceed disk quota: Space quota or files quota
 - Personal user directories and group-shared directories have a maximum available quota set
 - Delete and clean the files and try again
- Accidentally delete important files
 - File recovery is possible for /afs and /workfs2, with backups available for up to two weeks
 - /eos/user: valid for 3 days, and use the "eos recycle" command to restore files on your own
 - /home/lhaasso/user/.recyclebininternal: valid for 1 day, and restore files using the "cp" command on your own
- EOS file access slowly
 - Avoid issues like "eos ls" to the directory with large number of files



Ask for Help



群聊: LHAASO 计算平台用户技术交流群



该二维码7天内(8月11日前)有效, 重新进入将更新

- Three ways
 - Tel. (Working hour) : 010-88236855
 - **Email:** helpdesk@ihep.ac.cn (recommand)
 - Web page: <http://helpdesk.ihep.ac.cn>
 - Wechat
- It's better to give the information when ask for help
 - User name
 - Command, error message etc.
 - Better to provide detailed job info such as **JobID, submission time, job log, job path** etc. The more detailed, the easier for system manager to trace the error
- To feed back to job running slowly, please provide:
 - Which login node(run "hostname"), which directory or files the job access etc.
- User manual: <http://afsapply.ihep.ac.cn/cchelp/zh/>

Thank You!
Question?



IHEP School of Computing 2024 is coming!

- IHEP School of computing 2024 will be held in **Yanqing, Beijing** from the 21th to the 24th of August 2024
- 2.5 days, 21 lectures, and 4 hours of hands-on
- Indico: <https://indico.ihep.ac.cn/event/22917/>
- The course covers
 - Data processing in the field of high-energy physics,
 - AI technology for high-energy physics,
 - Computing technology for high-energy physics
 - Hands-on practice on computational platform



高能物理计算暑期学校
IHEP School of Computing

高能物理 第五届
IHEP School of Computing 2024
计算暑期学校

2024.8.21-8.24 北京·延庆

» 针对人群
暑期学校主要针对从事高能物理及其他学科的高年级本科生、研究生、博士后和科研工作者

» 培训内容
主要授课内容涉及主要授课内容涉及高能物理领域数据处理、高能物理AI技术、高能物理计算技术、计算平台上机实践等。

» 会议网址
<https://indico.ihep.ac.cn/e/isc2024>



请扫描上方二维码注册

» 联系方式
010-88236853; 13811502631 (晋老师)
010-88236883; 13811338941 (李老师)
Email: ComputingSchool@ihep.ac.cn