



2026

高能人工智能平台AI服务

Yiyu Zhang

zhangyiyu@ihep.ac.cn

Computing Center, IHEP, CAS

Outline



- 高能物理AI平台
- AI算力使用
 - 账号申请
 - 集群登录
 - 资源申请与使用
- 支持与服务
- 总结与未来计划



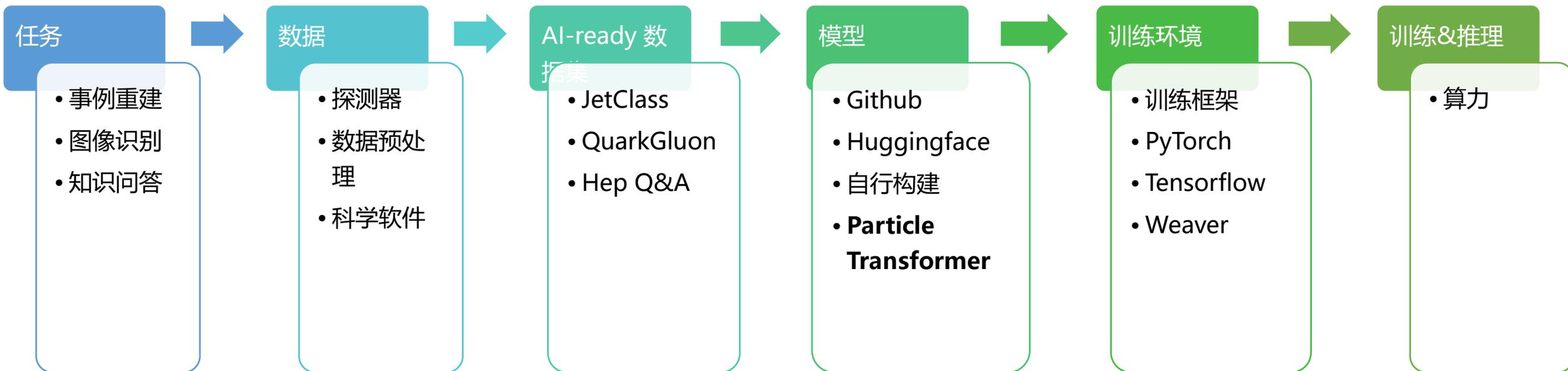
01

高能人工智能平台

背景

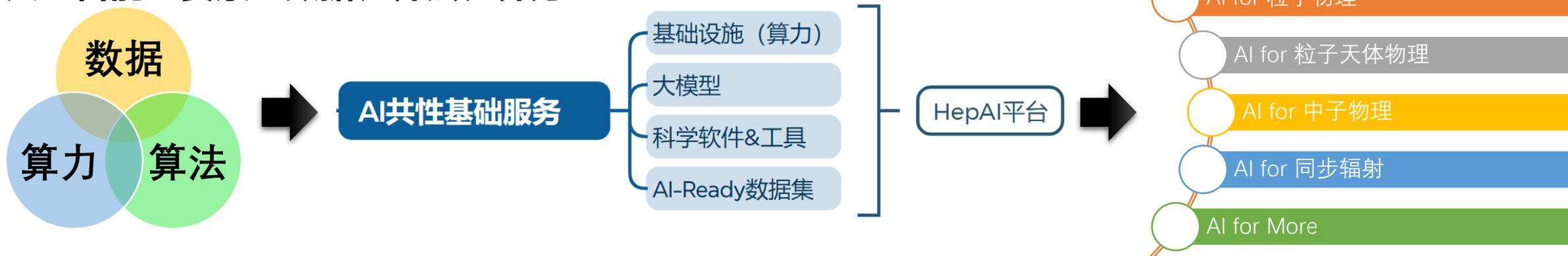


- AI研究流程：复杂、零碎，需要共性基础服务简化流程



- 目前高能所算力硬件共250+英伟达GPU，以V100为主，少量A100（非AI研究独占）。

- 人工智能三要素：数据、算法、算力



HAI高能人工智能平台



多学科场景的AI基础设施与协同引擎

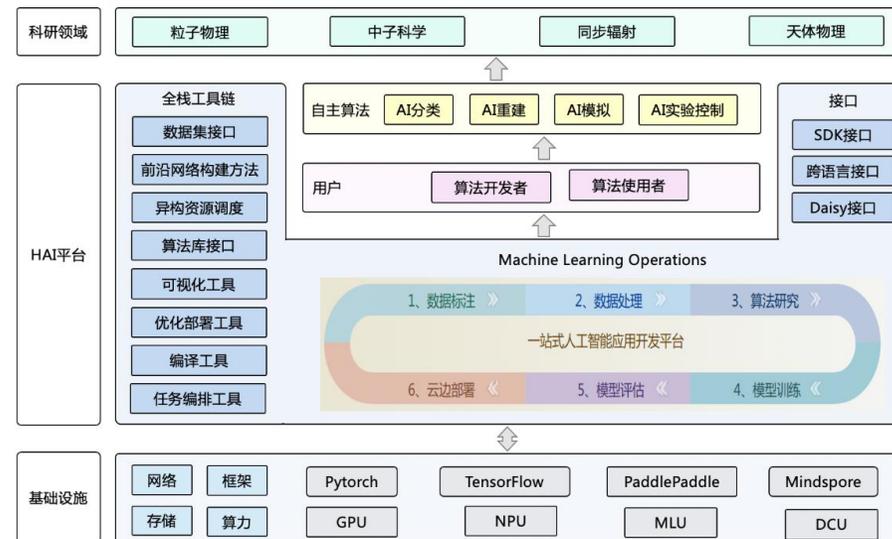
核心定位

☑ **共性基础设施**：面向粒子物理、粒子天体物理、同步辐射、中子科学、加速器等多领域，提供从数据治理、算法迭代到算力调度的一体化AI研发闭环。

☑ **软件定义平台**：以AI模型流为核心，打通“数据-算法-算力-协作”全链路，降低跨学科AI应用门槛。

平台价值

- **承载AI模型/工具**：已支持领域10+模型/工具。
- **新建专用AI算力 (2024)**：5PFlops (16卡GPU+ 48卡DCU), 200TB全闪存硬件
- **打通数据通道**：文本、语料、科学等10+AI数据集
- **赋能智能应用**：6+AI应用。



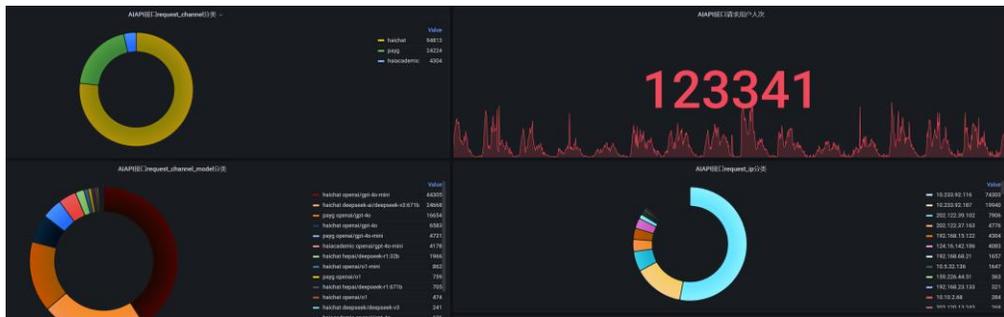
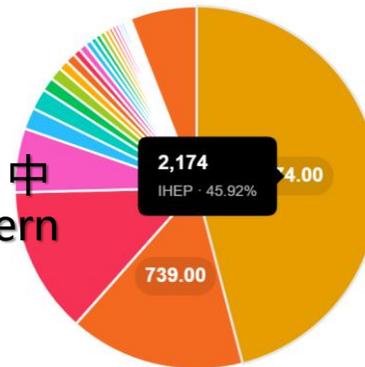
HepAI平台架构

□ **用户总数: 4850人**

◆ **高能所: 2174 (45%)**

◆ **中科大、国科大、中山大学, 北大, Cern**

□ **日均活跃: 800人**



过去15天, 模型请求人次10万+

HAI高能人工智能平台



AI平台不是纯硬件平台，它本身是软件系统，承载AI算法模型，打通数据通道，提供AI算力。

AI
共
性
支
撑
能
力

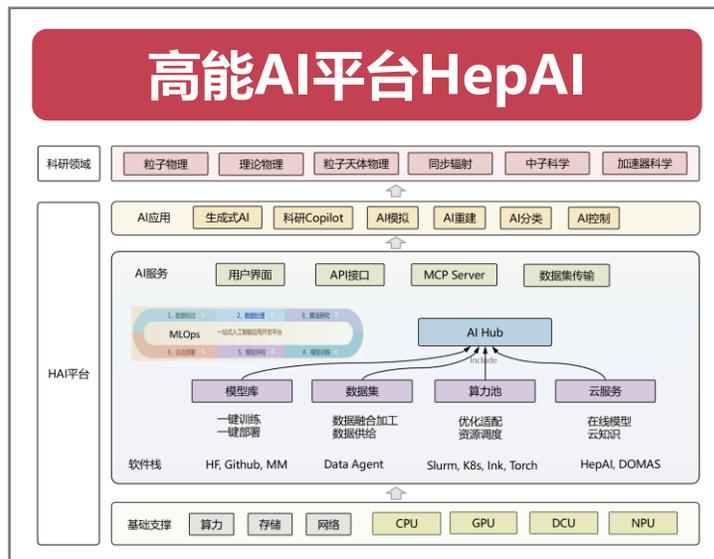
01 基础设施（算力）

02 大模型

03 科学软件&工具

04 AI-Ready数据集

整合



🚀 平台价值

- 对内提供公共服务，降低AI门槛
- 对外形成交流渠道，提升影响力

(1) 公共模型服务

- 大语言模型：DeepSeek, etc
- AI算法模型：SAM, ParT etc
- 科研工具：PDF解析 etc
- 一键部署、一键训练

(2) AI应用服务

- 生成式AI：HaiChat
- 学术助手：HaiAcademic(姜)
- 论文写作助手：Overleaf(侯)
- 快速开发/集成

(3) 公共算力服务

- 5PFlops, 200T SSD
- 领域模型在国产设备的适配
- AI易用（像私有服务器一样）
- 高效（智能调度提升整体利用率）

(4)
公共
数据
集
服
务

AI计算资源



- **8台服务器: 8*NV L40, 8*NV A800, 24*NV RTX5090, 48*DCU K100ai**

用途	设备名称	数量 (台)	配置
计算	A800 GPU服务器	1台	<ul style="list-style-type: none">• 8 * A800 80GB PCI-e NVIDIA GPU卡• 2 * Intel(R) Xeon(R) Gold 6430(32 core)• 1TB 内存, 7.68 TB NVME 本地硬盘
	L40 GPU服务器	1台	<ul style="list-style-type: none">• 8 * L40 48GB PCI-e NVIDIA GPU卡• 2 * Intel(R) Xeon(R) Gold 6430(32 core)• 1TB 内存, 7.68 TB NVME 本地硬盘
	RTX5090 GPU服务器	3台	<ul style="list-style-type: none">• 8 * RTX5090 32GB PCI-e NVIDIA GPU卡• 2 * Intel(R) Xeon(R) Gold 6430(32 core)• 500G 内存, 7.68 TB NVME 本地硬盘
	DCU服务器	6台	<ul style="list-style-type: none">• 8 * K100AI 64GB PCI-e 国产海光DCU卡• 2 * Intel(R) Xeon(R) Gold 6430(32 core)• 1TB 内存, 7.68 TB NVME 本地硬盘



02

AI算力使用

Hai平台主页: ai.ihep.ac.cn



- Ai 平台总入口
- 用户中心
 - Api key管理
 - 个人模型管理
 - 个人数据管理
 - 个人算力管理

- 模型
- 数据集
- 算力
- A应用

- 文档

账号申请



- 使用统一认证账号可登陆AI平台
- 使用AI平台算力服务需要以下条件

账号 - 高能AI平台手册

(<https://ai.ihep.ac.cn/hai-docs/getting-started/>)

(<https://ai.ihep.ac.cn/hai-docs/account/account/>)

算力管理

集群账号错误

Error listing jobs: 403: Error: User with email ihep_computing_service@ihep.ac.cn not found in the cluster.

无法访问集群资源

使用 AI 平台的完整功能需要满足以下条件:

前置要求:

1. 拥有高能所计算集群账号
2. 被添加到 IHEPAI 用户组
(详细说明请参照: <https://ai.ihep.ac.cn/hai-docs/account/account/>)

操作步骤:

- 第一步: 申请集群账号 (如未拥有)**
- 申请指南: <http://afsapply.ihep.ac.cn/cchelp/zh/accounts/>
 - 通常需要 1-2 个工作日审核

- 第二步: 申请 IHEPAI 用户组**
- 申请地址: <http://ccsuser.ihep.ac.cn>
 - 选择 "IHEPAI" 组并提交申请
 - 通常需要 2-3 个工作日审核

****注意:** 组权限变更通常需要 2-3 小时才能在系统中生效**

**仍有问题? **

- 邮箱: hepai@ihep.ac.cn
- 抄送: 系统管理员
- 请附上您的用户名和具体错误信息

请确保您已经配置了高能所集群账号。如果您还没有账号,请联系管理员申请。

[查看账号配置文档](#)

快速开始

高能AI算力平台

快速使用算力资源

申请账号

- [点击此处申请账号](#)

登录平台

- 登录AI平台 `ssh <username>@ailogin.ihep.ac.cn`

查看资源

- `sinfo` 查看集群资源情况, 详见[计算资源](#)
- 访问 ai.ihep.ac.cn/#/computing 查看计算资源

使用算力

您可以使用弹性算力服务或提交slurm作业使用算力资源。

启动弹性算力服务

- `hai-ecs` 启动弹性算力服务, 详见[AI弹性服务](#)
- `hai-ecs stop` 关闭弹性算力服务



目录

- 快速使用算力资源
- 申请账号
- 登录平台
- 查看资源
- 使用算力
 - 启动弹性算力服务
 - 启动slurm作业
- 快速使用模型服务
- 快速使用数据集服务

账号申请

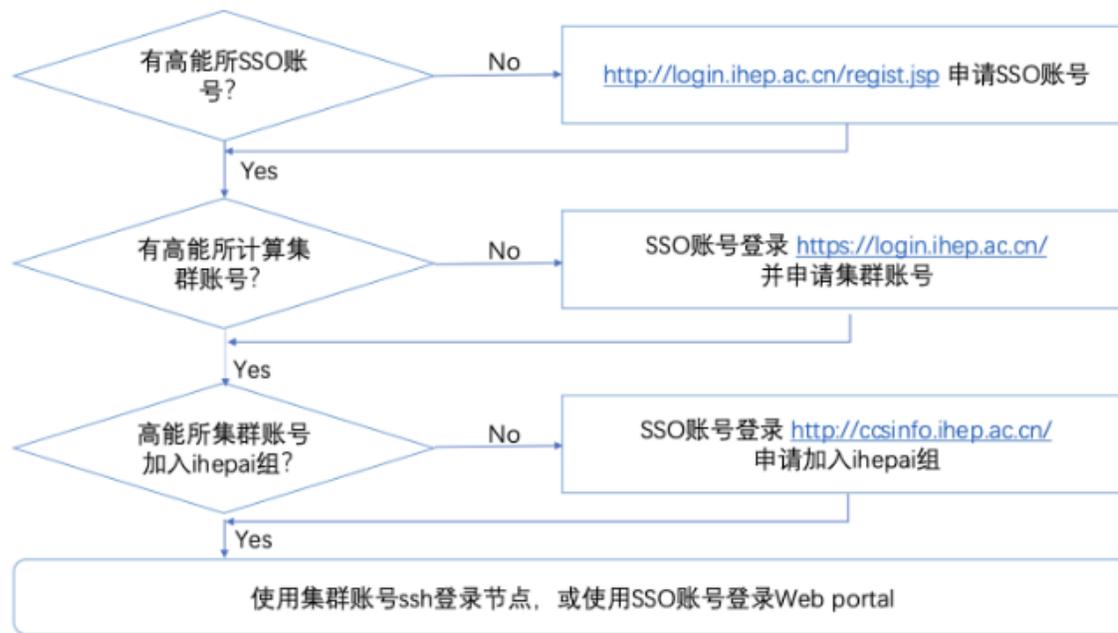


- 1. 如无高能所统一认证账号 (SSO), 点击[此处申请](#)
- 2. 如有高能所统一认证账号 (SSO), 但无高能所计算集群账号 (AFS), 点击[此处申请](#), 操作方法:
 - 登录后, 点击 [申请高能所计算平台账号](#), 填写信息相关信息
 - 申请时隶属应用可以选择 IHEPAI, 用户组选择 ihepai, 选择该组后, 会自动加入 IHEPAI 组, 无需第三步操作。
- 3. 如图:

- 3. 如先前已有统一认证 SSO 和计算集群账号 AFS, 但无未加入 IHEPAI 组, 点击[此处申请](#), 操作方法:
 - 登录后点击 [Apply to second linux group](#)
 - 点击 [Secondary group apply](#), 搜索 IHEPAI, 选择 IHEPAI, 点击 [确定](#)

审核后相关信息将通过邮件通知, 请注意查收。

等待30-60分钟系统信息同步后, 即可使用ssh方式登录 ailogin.ihep.ac.cn



experiment the user belonged to	linux gr
IHEPAI	ihepai

AI算力服务



- AI算力服务
 - 算力总览
 - 算力详细信息
- 个人算力统计
- 算力使用方法
 - 弹性算力使用

算力管理

刷新

总览

算力列表

个人算力

使用方法

SSH

ECS

EShell

JupyterLab

RemoteDesk

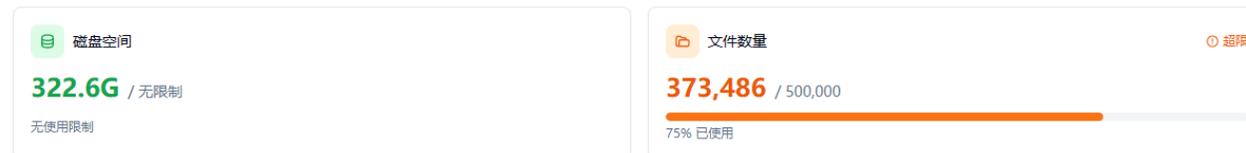
算力统计



使用统计 (2025.1.1-至今)



/aifs/user/home/zhangyiyu



/aifs/user/data/zhangyiyu

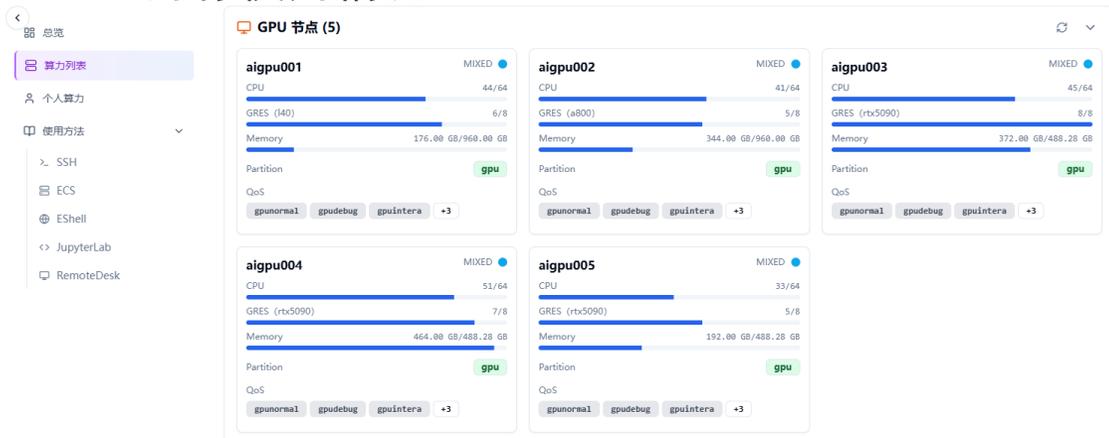


便捷使用



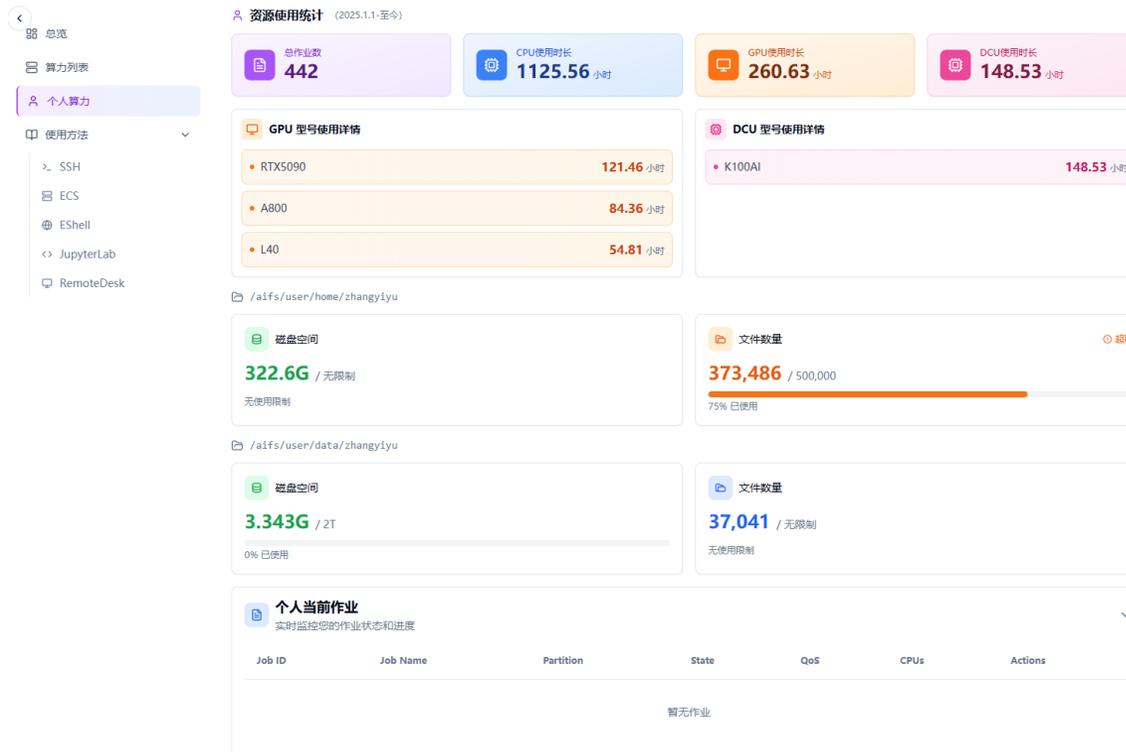


- 算力详细信息
 - 每个算力节点的CPU GPU 内存使用情况



- 算力使用方法
 - SSH
 - ECS 弹性算力服务
 - Eshell 网页shell服务
 - Jupyter网页服务
 - Remote Desk 远程桌面服务

- 个人算力统计
 - 总作业数、CPU、GPU等使用情况
 - 每个型号AI加速卡的使用情况
 - 个人存储使用情况

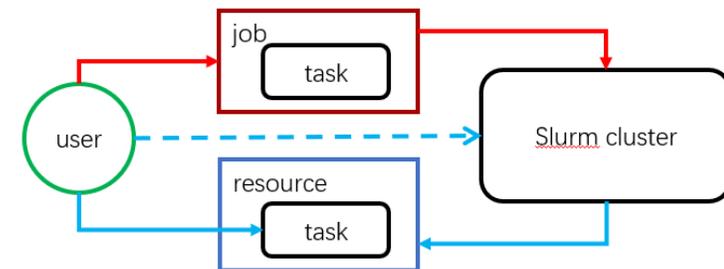


ECS 弹性算力服务



ECS 弹性算力服务

- 弹性云服务器 (Elastic Cloud Server, ECS) 为用户提供可直连、可扩展的算力服务器
- 允许用户动态调整计算资源的配置, 包括CPU、内存、GPU卡等, 满足灵活的AI算力需求
- 主要用于单机单卡、单机多卡的AI模型调试、训练和部署
- 允许用户像使用个人服务器一样直接与算力资源进行交互
- 优势, 即支持对AI模型和软件代码的运行时调试



弹性算力服务 (hai-ecs)

使用 hai-ecs 工具快速获取弹性算力资源

web启动ecs hai-ecs 使用

The screenshot shows the 'Interactive Space' (交互式空间) section of the hai-ecs web interface. A red box highlights the configuration options, including hardware type (GPU), configuration (Nvidia L40), and duration (00:30:00). Below this, a list of available configurations is shown, with 'Nvidia L40 · 1 node · 1 card · 1 CPU · 24 GB' selected. The 'Running' (运行中) section shows '暂无运行中的空间' (No running spaces). The 'Other States' (其他状态) section shows three instances in various states (Public, New Space, etc.).

A screenshot of a running ECS instance. The status is '运行中' (Running) with a count of 1. The instance is named 'New Space' and has the configuration 'Nvidia RTX5090 · 1 node · 1 card · 1 CPU · 24 GB'. The user is 'zhangyiyu@ihep.ac.cn' and it was created 2 seconds ago.



Control buttons for ECS instances: 停止 (Stop), 编辑 (Edit), 启动 (Start), 删除 (Delete).

A screenshot of a terminal window showing the process of connecting to the ECS instance via SSH and setting up the environment. The terminal output includes the SSH command and the installation of the dcu environment using module and conda.

```
ssh -o UserKnownHostsFile=/dev/null zhangyiyu@ai.ihep.ac.cn -p 53307

使用ssh登陆到计算节点

可以使用module和conda配置dcu环境

使用公共环境

module load /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles/pytorch/2.3.0-dcu-py310

使用自己的conda

module use /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles
module avail
module load anaconda/24.3.0
conda create -n python310-torch230-dcu python=3.10 \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsrc/main \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsrc/r \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgsrc/msys2
conda activate python310-torch230-dcu
# 重要, 必须使用dcu的pytorch
pip install /aifs/public/soft/downloads/torch-2.3.0+das.opt2.dtk24043-cp310-cp310-manylinux_2_28_x86_64.whl -U
```

ECS 弹性算力服务



- 在AI集群上用命令行启动ECS

算力管理

- `hai-ecs`, 启动ECS, 约5秒钟, 启动后通过 `ssh <username>@<ip> -p <port>` 登录服务器, 需输入集群个人密码登录。
- 再次 `hai-ecs` 或使用 `hai-ecs status` 查看ECS状态, 包括 `ip`、`port` 等信息。
- `hai-ecs stop` 关闭ECS, 释放资源, 修改自动保存。
- `hai-ecs -h` 查看更多参数设置。
- `-g GRES`, `--gres GRES` 设置加速卡类型和卡数, 例如: `-g gpu:1` 表示1张GPU卡, `-g dcu:1` 表示1张DCU卡。
- `-t TIME`, `--time TIME` 设置机器的最大运行时间, 默认是 120m (分钟), 可以使用 `h` 表示小时, `d` 表示天。
- `-tp GPU_TYPE`, `--gpu-type GPU_TYPE` 设置GPU类型, 默认是 A800, 可选值包括 A800、L40、K100AI。

```
(base) [zhangyiyu@ailogin001 ~]$ hai-ecs
Applying Elastic Cloud Server (ECS) job with the following configuration:
  num_nodes      : 1
  cpu_cores      : 8
  memory         : 32
  accerlerator_cards: gpu * 1 (rtx5090)
Job submitted, job_id: `20681`
Waiting for the job to be running ... 0.26s.查询次数: 0, Job (id=`20681`) current status: RUNNING

The ECS is ready!
HostName ai.ihep.ac.cn
User zhangyiyu
Port 53307
UserKnownHostsFile /dev/null
StrictHostKeyChecking no

node: aigpu004
for aigpu[001-002]( L40, A800 ), please use cuda12;
for aigpu[001-002]( 5090 ), please use cuda13;
for aidcu , please use cuda for dcu;

conda public environment:
  cuda 12: /aifs/public/env/py10-gpu-root
  cuda 13: /aifs/public/env/py10-gpu-cuda13-root
  dcu : /aifs/public/env/py10-dcu-root
you also can use conda create command to create your env

For more information, please visit: `https://ai.ihep.ac.cn/#/docs`
You can connect to it via: `ssh -o UserKnownHostsFile=/dev/null zhangyiyu@ai.ihep.ac.cn -p 53307`
```

您也可以通过 `slurm` 命令直接管理ECS。+ `squeue` 查看作业队列情况, 获取 `job_id`, 如: `30` + `scancel <job_id>` 关闭虚拟机, 释放资源, 修改自动保存。



- 登录使用

通过 `ssh` 登录后，您将登录到分配了资源的计算节点服务器中。

- 用户的家目录与登录节点相同，均为 `/aifs/user/home/<username>`，
- `nvidia-smi` 查看GPU情况， `hy-smi` 查看DCU情况。
- `df -h` 查看磁盘使用情况。
- `htop` 查看CPU、内存、进程等信息。
- `source ~/.bashrc` 加载AI集群环境变量，如Conda等。

```
[zdzhang@ailogin001 ~]$ ssh -o UserKnownHostsFile=/dev/null zdzhang@ai.ihep.ac.cn -p 53081
Warning: Permanently added '[ai.ihep.ac.cn]:53081' (ED25519) to the list of known hosts.
zdzhang@ai.ihep.ac.cn's password:
[zdzhang@aigpu002 ~]$ source ~/.bashrc
(base) [zdzhang@aigpu002 ~]$
```

注： `hai-ecs v2` 基于slurm调度，启动速度快，但不再提供root权限，用户软件可自行安装在家目录中，如需系统层面的软件、库、依赖库等，请联系 `helpdesk.ihep.ac.cn`。

在VS code 中连接ECS服务

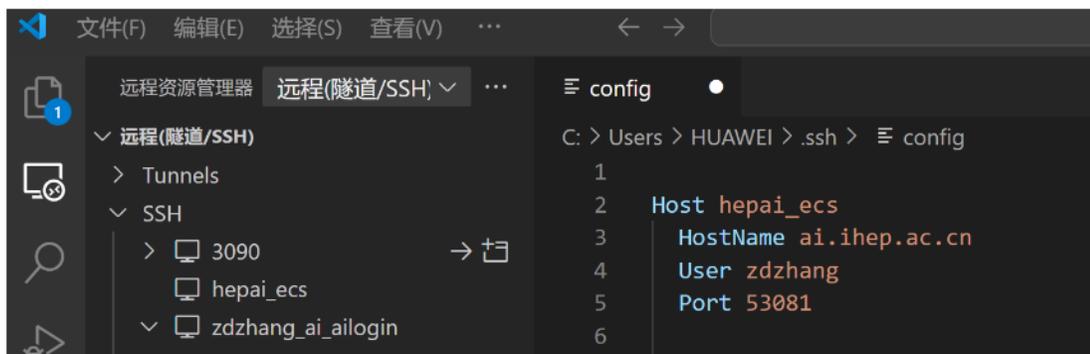


本地VSCode直连ECS

在本地VSCode安装 Remote - SSH 插件后，可以实现直连ECS。

点击 远程资源管理器 - SSH - 打开SSH配置文件，编辑 ~/.ssh/config 文件，添加ECS的连接信息。配置如下：

```
Host hepai_ecs
  HostName <ECS_IP>
  User <username>
  Port <ECS_PORT>
```



配置项说明：- hepai_ecs 为自定义远程服务器名，可更改 - HostName 为ECS的IP地址，为 ai.ihep.ac.cn - User 为AI集群用户名 - Port 为ECS的端口号，此处不是默认 22，需要从 hai-ecs 命令获取，不同用户端口号不同，每位用户端口号固定。

连接ECS：- 配置完成后，在 远程资源管理器 的 SSH 下点击刷新，找到 hepai_ecs，点击连接，输入密码，连接成功。

注：- ECS为内部服务器，需在内网环境下连接，在外网环境下需提前打开VPN：

vpn.ihep.ac.cn。

Salloc 无法实现同样的功能

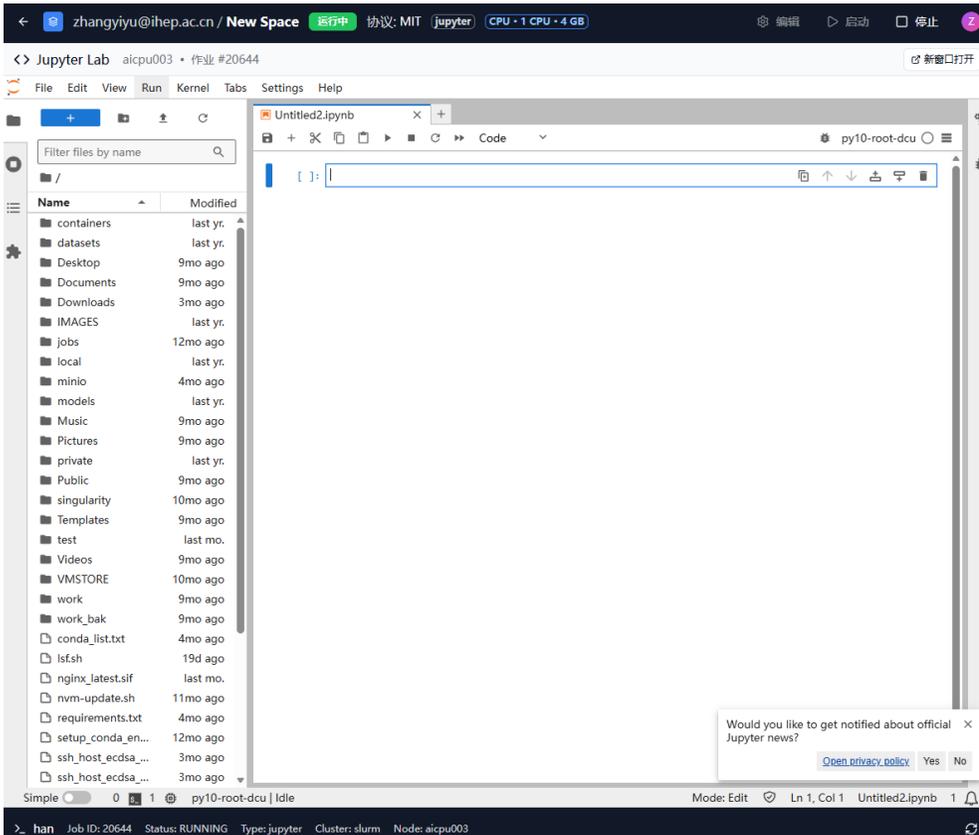
```
(base) [zhangyiyu@ailogin001 ~]$ salloc -p dcu -q dcunormal
salloc: Granted job allocation 2473
salloc: Nodes aidcu005 are ready for job
/aifs/user/home/zhangyiyu
(base) bash-5.1$
```

```
(python310-torch230-dcu) [zhangyiyu@ailogin001 slurm_2025070712_88b2fadb]$ ssh aidcu005
The authenticity of host 'aidcu005 (10.5.6.21)' can't be established.
ED25519 key fingerprint is SHA256:ee24zqarzDkt5c52WVGuNLBofrH4NZJCR/R+P4P8k7s.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'aidcu005' (ED25519) to the list of known hosts.
zhangyiyu@aidcu005: Permission denied (publickey,qssapi-keyex,qssapi-with-mic).
```

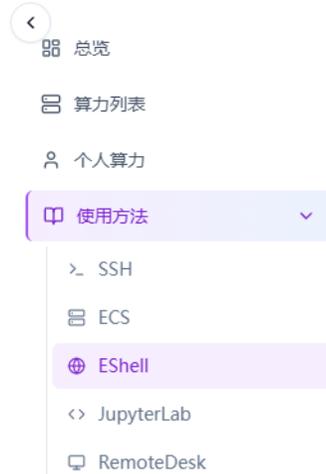
JupyterLAB 和 Eshell



- JupyterLAB适合在线查看、编辑文件...



- EShell 是一个基于 Web 的终端工具，可以直接在浏览器中访问集群终端，无需安装任何客户端软件。



EShell 终端

使用 EShell 在浏览器中直接访问集群终端

[返回教程](#)

[新窗口打开](#)

```
Permission denied, please try again.
zhangyiyu@ailogin001.ihep.ac.cn's password:
Permission denied, please try again.
zhangyiyu@ailogin001.ihep.ac.cn's password:
#####
Welcome to IHEP AI Platform
#
# - To list QOS, please use the slqos command.
# - To list association, please use the slas <username>
# command, where replace <username> with your username.
#
# Any Question, Please contact http://helpdesk.ihep.ac.cn/
#
#####
Last failed login: Tue Mar 24 21:58:00 CST 2026 from 10.5.6.18 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Tue Mar 24 17:05:58 2026 from 10.10.15.244
-----
Current storage usage:
Disk quotas for prj 112 (pid 112):
  Filesystem      used  quota  limit  grace  files  quota  limit  grace
/aifs/user/home/zhangyiyu
                    322.6G  0k     0k     -     373530 300000 500000  -
Disk quotas for prj 111 (pid 111):
  Filesystem      used  quota  limit  grace  files  quota  limit  grace
/aifs/user/data/zhangyiyu
                    3.343G  0k     2T     -     37042  0      0      -
-----
When exceeding the maximum storage quota, please clean up unnecessary files or contact the administrator to apply for an expansion of the quota.
(base) [zhangyiyu@ailogin001 ~]$ _
```

使用ssh方式登陆AI集群



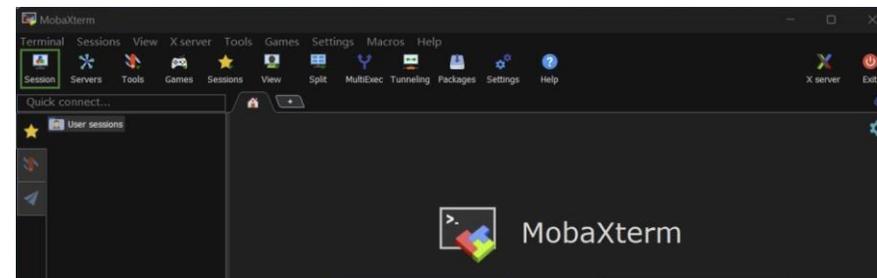
- Linux\Mac用户可通过命令行登录集群
 - ssh username@ailogin.ihep.ac.cn
- Windows用户则可以通过ssh客户端登录服务器
 - MobaXterm
 - <https://mobaxterm.mobatek.net>
 - Xshell
 - Cmd & PowerShell
 - Win+X -> 终端
 - Vs code

```
zhangyiyu@ailogin001:~  
Windows PowerShell  
版权所有 (C) Microsoft Corporation. 保留所有权利。  
安装最新的 PowerShell, 了解新功能和改进! https://aka.ms/PSWindows  
PS C:\Users\zhyiy> ssh zhangyiyu@ailogin.ihep.ac.cn  
#####  
#           Welcome to IHEP AI Platform           #  
# Any Question, Please contact http://helpdesk.ihep.ac.cn/ #  
#####  
Last login: Tue May 20 11:14:16 2025 from 192.168.32.125  
/aifs/user/home/zhangyiyu  
(base) [zhangyiyu@ailogin001 ~]$
```

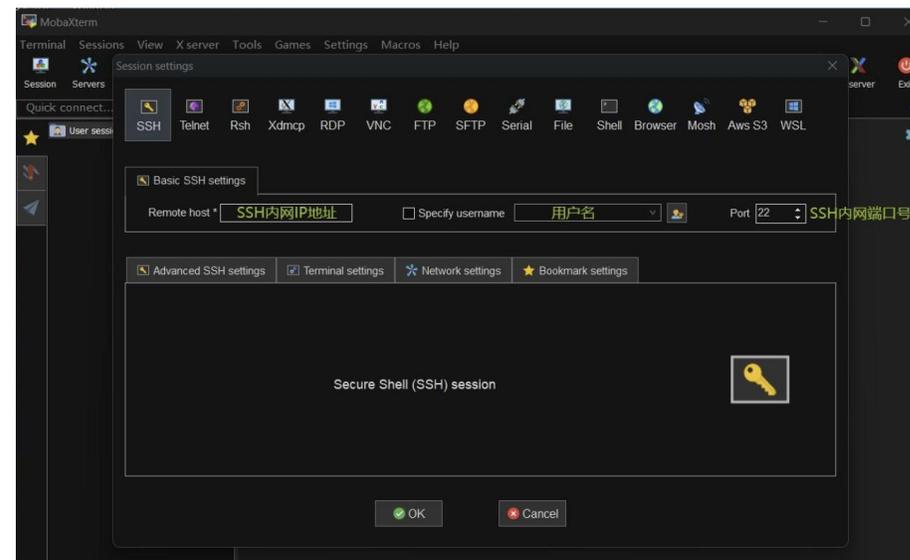


MobaXterm

点击绿色方框处:



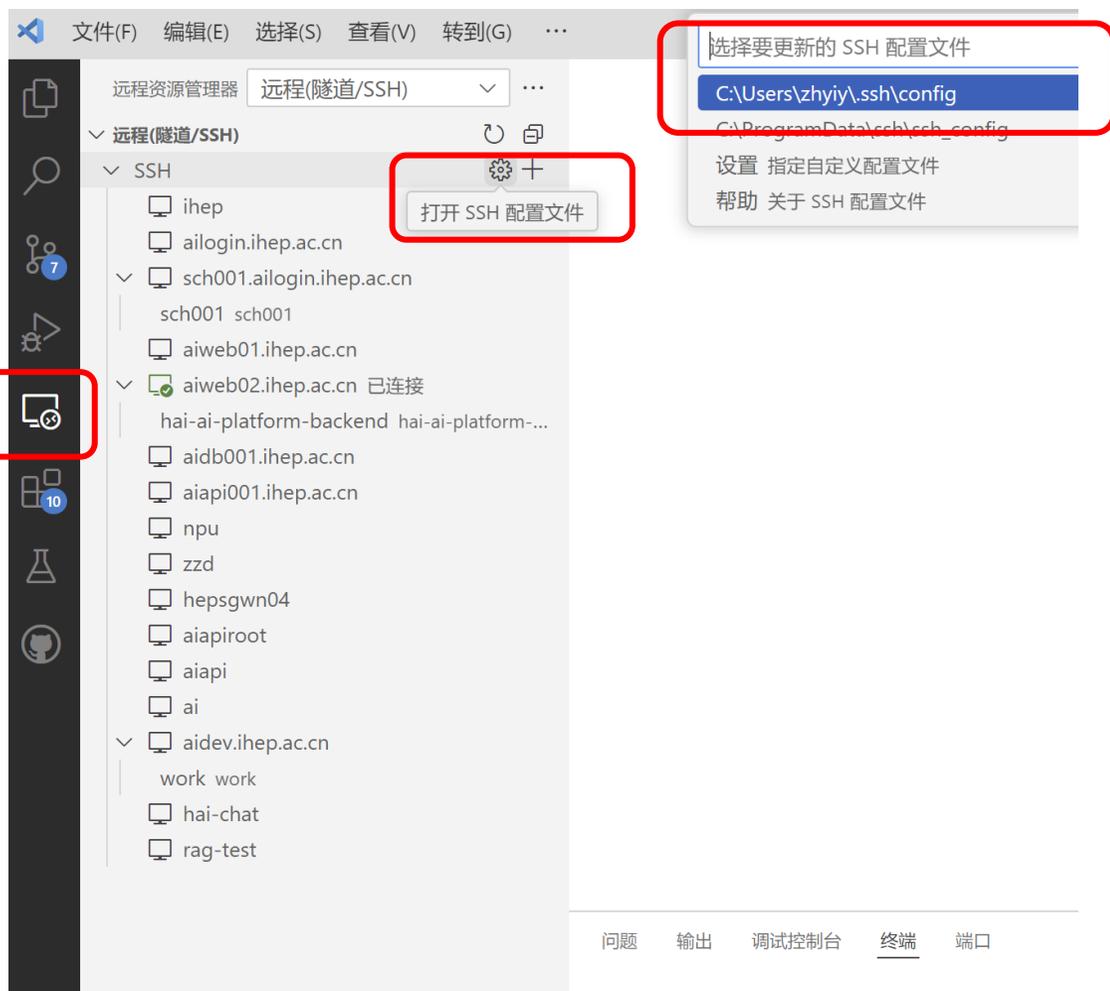
依次填入网址、用户名和端口号 (22)



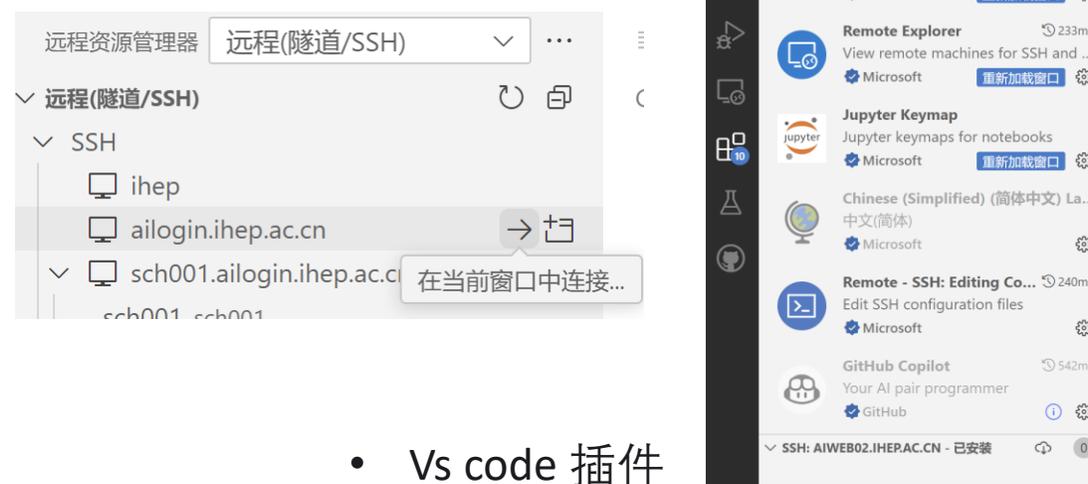
使用ssh方式登陆AI集群



- VS code
 - <https://code.visualstudio.com/>



```
config
C: > Users > zhyiy > .ssh > config
1 # Read more about SSH config files: https://linux.die.net/man/5/
2 Host ailogin.ihep.ac.cn
3     HostName ailogin.ihep.ac.cn
4     User zhangyiyu
5     IdentityFile C:\Users\zhyiy\.ssh\id_rsa
6     ForwardAgent yes
7     ForwardX11 yes
8     ForwardX11Trusted yes
9
```



- Vs code 插件

AI 集群环境



- 加载公共环境

- 如果缺少软件包，请联系管理员，或使用自己的python环境

- 创建自己的环境

在terminal输入

```
module purge
module use /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles
module load anaconda/24.3.0
source
/cvmfs/slurm.ihep.ac.cn/alma9/anaconda3/bin/activate
```

- for aigpu[001-002](L40, A800), please use **cuda12**;
- for aigpu[003-006](5090), please use **cuda13**;
- for aidcu(k100ai), please use **cuda for dcu**;
- conda public environment:
 - **cuda 12**: /aifs/public/env/py10-gpu-root
 - **cuda 13**: /aifs/public/env/py10-gpu-cuda13-root
 - **dcu** : /aifs/public/env/py10-dcu-root

使用自己的conda

```
module use /cvmfs/slurm.ihep.ac.cn/alma9/modulefiles
module avail
module load anaconda/24.3.0
conda create -n python310-torch230-dcu python=3.10 \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/main \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/r \
-c https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/msys2
conda activate python310-torch230-dcu
# 重要, 必须使用dcu的pytorch
pip install /aifs/public/soft/downloads/torch-2.3.0+das.opt2.dtk24043-cp310-cp310-
manylinux_2_28_x86_64.whl -U
```

光合开发者社区 > pytorch > DAS1.3

DCU 使用:

必须使用基于dtk-24.04的软件包

[首页 | 光合开发者社区](#)

<https://cancon.hpccube.com:65024/4/main/>

文件名

←	pytorch
<input type="checkbox"/>	torch-2.1.0+das.opt2.dtk24043-cp38-cp38-manylin
<input type="checkbox"/>	torch-2.1.0+das.opt2.dtk24043-cp310-cp310-many
<input type="checkbox"/>	torch-2.1.0+das.opt2.dtk24043-cp311-cp311-many
<input type="checkbox"/>	torch-2.3.0+das.opt2.dtk24043-cp38-cp38-manylin
<input type="checkbox"/>	torch-2.3.0+das.opt2.dtk24043-cp310-cp310-many
<input type="checkbox"/>	torch-2.3.0+das.opt2.dtk24043-cp311-cp311-many



Setup

AI平台文档说明

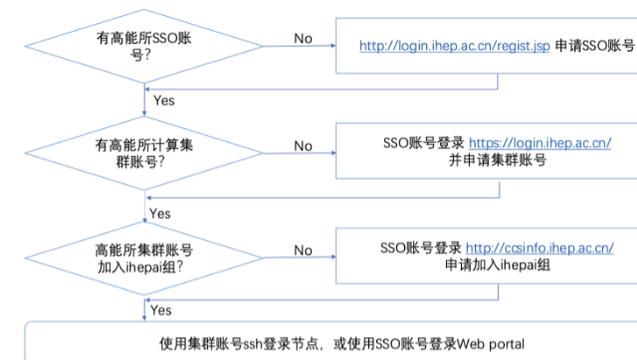
Site structure

文档包括以下几个主要部分:

Model - AI模型服务, 包括模型介绍、使用说明等	Dataset - AI数据集服务, 包括数据集介绍、使用说明等
Computing - AI算力服务, 包括算力介绍、使用说明等	Application - AI应用服务, 包括应用介绍、使用说明等
Docs Info - AI文档说明, 包括如何编写和发布文档	Developer - 开发者指南, 包括 Dr.Sai 智能体的开发说明
Help - 帮助	

目录
Site structure

申请账号



- 1.如无高能所统一认证账号 (SSO) , 点击[此处申请](#)
- 2.如有高能所统一认证账号 (SSO) , 但无高能所计算集群账号 (AFS) , 点击[此处申请](#), 操作方法:
 - 登录后, 点击 [申请高能所计算平台账号](#), 填写信息相关信息
 - 申请时隶属应用可以选择 IHEPAI, 用户组选择 ihapai, 选择该组后, 会自动加入 IHEPAI 组, 无需第三步操作。
- 如图:





Slurm (Simple Linux Utility for Resource Management)

是一种开源的作业调度和资源管理系统，广泛应用于超级计算中心和高性能计算（HPC）集群中。Slurm的主要功能包括资源分配、任务管理和作业调度，旨在优化并行计算任务的处理效率和资源利用率。

任务->调度系统->计算节点

1. 编写作业脚本

```
vi test.slurm # 根据需求，选择计算资源：CPU 或 GPU、所需核数、是否需要大内存
```

2. 提交作业

```
sbatch test.slurm
```

示例 SLURM 脚本

以下是一个简单的 SLURM 作业脚本示例：

```
#!/bin/bash
#SBATCH --job-name=test_job # 作业名称
#SBATCH --partition=dcu # 队列名称
#SBATCH --ntasks=1 # 总任务数 = 总 CPU 数
#SBATCH --gres=dcu:1 # 使用的 dcu 数量
#SBATCH --time=01:00:00 # 作业运行时间限制
#SBATCH --output=test_job.out # 标准输出文件
#SBATCH --error=test_job.err # 错误输出文件

# 加载必要的模块
module load anaconda/24.3.0
module load cuda/11.3

# 激活环境
source activate my_env

# 运行命令
python my_script.py
```

• 资源管理：按功能分区

```
control up infinite 1 drain aislurm01
login up infinite 1 drain ailogin001
gpu up infinite 1 drng aigpu002
gpu up infinite 1 mix aigpu001
dcu up infinite 1 mix aidcu006
dcu up infinite 5 idle aidcu[001-005]
```

- 算力分区
- 根据加速卡类型划分
- dcu算力分区和gpu算力分区

• 作业调度：基于QOS的优先级调度，作业中指定QOS即可获得对应的调度服务

QOS名称

作业优先级

作业可使用的最大资源限制

作业最大运行时间

可提交的最大作业数量

Name	Priority	MaxTRESPU	MaxWall	MaxSubmitPU
dcudebug	20	cpu=32,gres/dcu:k100ai=4,mem=960G	00:15:00	24
dcunormal	0	cpu=16,gres/dcu:k100ai=2,mem=480G	2-00:00:00	16
gpudebug	20	cpu=16,gres/gpu:a800=4,gres/gpu:l40=4,gres/gpu=4,mem=960G	00:15:00	8
gpunormal	0	cpu=8,gres/gpu:a800=2,gres/gpu:l40=2,gres/gpu=2,mem=480G	2-00:00:00	4

3.2.2 Slurm 作业

(<http://afsapply.ihep.ac.cn/cchelp/zh/local-cluster/jobs/slurm/>)

AI模型服务: AI models and tools



- Use API-KEY to directly call AI models
- Get api-key from ai.ihep.ac.cn

HepAI 模型 数据集 算力 应用 文档手册 EN 2

个人中心
管理您的账户、API密钥、用量统计等

概览

欢迎回来, zhangyiyu@ihep.ac.cn!
这是您的个人中心概览, 快速查看账户状态和最近调用

账户余额: ¥313.87 (约 78,466,913.5 tokens)

API密钥: 5 (3个活跃)

今日调用: 10 (+100%)

待审核: 0 (需要处理)

最近调用

操作	模型	时间	费用
调用	hepai/bge-m3:latest	1小时前	-¥0.00
调用	hepai/bge-m3:latest	1小时前	-¥0.00
调用	hepai/bge-m3:latest	1小时前	-¥0.00
调用	hepai/bge-m3:latest	1小时前	-¥0.00
调用	hepai/bge-m3:latest	1小时前	-¥0.00

快速操作

- 充值
- 创建密钥
- 查看用量
- 管理课题

账户管理
管理您的账户余额和充值记录

账户余额: ¥313.87 (约 78,466,914 tokens)

free

+ 充值

升级Plus

API密钥管理
管理您的API密钥, 用于程序化访问平台服务

使用步骤

- 复制 api_key
- 根据接口文档调用

注意 base_url 地址: <https://aiapi.ihep.ac.cn/api/v2>

安全提示

请妥善保管您的API密钥, 不要在公共场所或代码仓库中暴露。如果密钥泄露, 请立即删除并重新生成。

+ 创建新密钥

我的API密钥

序号	名称	密钥	过期时间	备注	操作
1	test	sk-EL*****qp8nm	2027-06-04	-	删除
2	test	sk-Qo*****jgfhg	2030-09-06	-	删除

AI模型服务: AI models and tools



- View API-KEY available models on the website
- List available models through API

- `curl`

<https://aiapi001.ihep.ac.cn/apiv2/v1/models> -H "authorization: Bearer sk-xxx"

HepAI: 本地部署的模型

The screenshot shows the HepAI website interface. On the left, there is a sidebar with navigation options: Tasks, Libraries, Providers, Apps, and Inference Providers. The main content area displays a list of AI models. The models listed are:

- bge-m3:latest** by wanghong@ihep.ac.cn. Description: 演示用模型, 用于测试和体验. Provider: HepAI. Status: 云模型. Updated 3 months ago.
- qwen3_30b** by wanghong@ihep.ac.cn. Description: 阿里的Qwen模型. Provider: HepAI. Status: 云模型. Updated 5 months ago.
- deepseek-v3.2** by wanghong@ihep.ac.cn. Description: 演示用模型, 用于测试和体验. Provider: HepAI. Status: 云模型. Updated 1 month ago.
- bge-reranker-v2-m3** by wanghong@ihep.ac.cn. Description: 演示用模型, 用于测试和体验. Provider: HepAI. Status: 云模型. Updated 3 months ago.
- bge-m3-hybrid** by wanghong@ihep.ac.cn. Description: 演示用模型, 用于测试和体验. Provider: HepAI. Status: 云模型. Updated 15 days ago.

The screenshot shows the HepAI website interface, specifically the '模型广场' (Model Plaza) section. The page displays a grid of AI models with details such as model ID, provider, description, and update status. The models shown are:

- gpt-5.2-pro** by admin. Description: gpt-5.2-pro模型. Provider: OpenAI. Status: 云模型. Updated 4 days ago.
- glm-4.7** by admin. Description: 智谱清言GLM系列模型. Provider: Zhipu. Status: 云模型. Updated 4 days ago.
- claude-sonnet-4-6** by admin. Description: 安全可靠的AI助手, 擅长对话和分析. Provider: Anthropic. Status: 云模型. Updated 4 days ago.
- claude-sonnet-4-5-20250929-thinking** by admin. Description: 安全可靠的AI助手, 擅长对话和分析. Provider: Anthropic. Status: 云模型. Updated 4 days ago.
- minimax-m2.7-highspeed** by admin. Description: 专业AI模型, 提供优质服务. Provider: Minimax. Status: 云模型. Updated 4 days ago.
- claude-haiku-4-5-20251001-thinking** by admin. Description: 安全可靠的AI助手, 擅长对话和分析. Provider: Anthropic. Status: 云模型. Updated 4 days ago.
- claude-opus-4-5** by admin. Description: 安全可靠的AI助手, 擅长对话和分析. Provider: Anthropic. Status: 云模型. Updated 4 days ago.
- claude-sonnet-4-5** by admin. Description: 安全可靠的AI助手, 擅长对话和分析. Provider: Anthropic. Status: 云模型. Updated 4 days ago.
- s1-base-lite** by hepai. Description: 专业AI模型, 提供优质服务. Provider: Scienceone. Status: 云模型. Updated 2 months ago.
- gpt-5.4** by admin. Description: 强大的通用语言模型, 适合各种文本生成任务. Provider: OpenAI. Status: 云模型. Updated 4 days ago.

AI模型服务: AI models and tools



- 示例:
 - 遍历当前文件夹所有.md文件, 使用llm为每个文件内容生成摘要。

```
import os
from hepai import HepAI
from typing import Dict, List

client = HepAI(
    api_key=os.environ.get("HEPAI_API_KEY"),
    base_url="https://aiapi.ihep.ac.cn/apiv2",
)
```

```
def main():
    # 调用函数处理当前文件夹所有.md文件
    result = summarize_md_files()

    # 格式化输出结果
    output_lines = []
    for filename, summary in result.items():
        output_lines.append(f"文件: {filename}")
        output_lines.append(f"摘要: {summary}")
        output_lines.append("-" * 50)

    # 将结果拼接为字符串返回
    return "\n".join(output_lines)
```

```
def summarize_md_files(model: str = "hepai/deepseek-v3.2", max_tokens: int = 150) -> Dict[str, str]:
    """
    遍历当前文件夹所有.md文件, 使用llm为每个文件内容生成摘要。

    Returns:
        Dict[str, str]: 一个字典, 键为.md文件名, 值为该文件的摘要内容。
        如果处理过程中发生错误, 摘要值将为错误信息字符串。
    """

    # 遍历每个.md文件
    for md_file in md_files:
        file_path = os.path.join(current_dir, md_file)

        try:
            # 读取文件内容
            with open(file_path, 'r', encoding='utf-8') as f:
                content = f.read()

            # 构建提示词
            prompt = f"请为以下Markdown文档内容生成一个简洁的摘要, 摘要长度不超过{max_tokens}个token: \n\n{content}"

            # 调用LLM API
            response = client.chat.completions.create(
                model=model,
                messages=[
                    {"role": "system", "content": "你是一个专业的文档摘要助手。"},
                    {"role": "user", "content": prompt}
                ],
                max_tokens=max_tokens,
                temperature=0.5
            )

            # 提取摘要
            summary = response.choices[0].message.content.strip()
            summaries[md_file] = summary

    return summaries
```

AI应用: HaiChat



deepseek-ai/deepseek-r1:671b

Q 搜索模型

- deepseek-ai/deepseek-v3:671b 高并发 满血版
- deepseek-ai/deepseek-r1:671b 高并发 满血版 深度思考 ✓
- deepseek-ai/deepseek-r1:32b 高并发 蒸馏版 深度思考
- deepseek-ai/deepseek-r1:7b 高并发 蒸馏版 深度思考
- hepai/deepseek-r1:671b 本地模型 满血版 深度思考
- hepai/deepseek-r1:32b 本地模型 蒸馏版 深度思考
- openai/gpt-4o 高并发

建议

Help me study
vocabulary for a college entrance exam

Give me ideas
for what to do with my kids' art

Overcome procrastination
give me tips

deepseek-ai/deepseek-r1:671b

- Multi-model comparison
- Real-time message modification
- Regenerate: (Regenerate, enrich, refine)
- Continue generating

<https://chat.ihep.ac.cn>

openai/gpt-4o-mini

```
openai/gpt-4o
# "http://192.168.15.237:42701/v1/chat/completions",
# headers={"Authorization": f"Bearer {hepai_api_key}"},
# json=data,
# stream=True,
# ) 可以访问, 但aiohttp.ClientSession r = await session.request 失败, 为什么
```

openai/gpt-4o-mini

如果您能够使用 `requests.post` 成功访问 API, 但在使用 `aiohttp.ClientSession` 时遇到问题, 可能是因为以下一些原因。我们可以逐一检查这些可能性, 以解决您面临的问题。

1. 请求头问题

确保使用 `aiohttp` 时您正确设置了所有请求头, 包括 `Authorization`。例如:

```
python
import aiohttp
```

openai/gpt-4o

当您使用 `requests.post` 可以成功访问, 而使用 `aiohttp.ClientSession` 的 `await session.request` 失败时, 可能的原因有以下几种:

1. Session 配置:
 - 确保 `aiohttp.ClientSession` 的配置和 `requests.post` 相符。特别注意请求头、认证信息和其他参数。
2. Event Loop:
 - 确保你的代码在适当的事件循环中运行。 `aiohttp` 基于异步 I/O, 确保

+ 输入消息

大语言模型可能会生成误导性错误信息。请对关键信息加以验证。

All users: 6016 (IHEP 2980)
Active users: 923 (within 1 month)

AI应用：其他



应用

探索我们的智能工具集合，为您的研究和工作提供强大的支持

C

ChatBox

by Zhengde Zhang et al.

Chatbox AI 是一款 AI 客户端应用和智能助手，支持众多先进的 AI 模型和 API，可在 Windows、MacOS、Android...

打开 →

H

HaiChat

by Yiyu Zhang et al.

基于大语言模型的智能问答平台，支持多轮对话与专业领域问答

打开 →

D

Dr.Sai 智能体

by Zhengde Zhang, Yiyu Zhang, Bolun Zhang...

高能物理领域的智能问答助手

打开 →

H

HaiAcademic GPTA

by Fayu Jiang et al.

学术研究辅助工具，支持文献综述与论文写作

打开 →

P

PDF解析器

高效的PDF文档内容提取与解析工具

打开 →

X

XRD强度二维映射应用

by Qingmeng Li et al.

对批量实验图像自动积分、计算区间强度值、根据样品实验排布顺序进行强度值mapping

打开 →

X

XRD寻峰扣背底应用

by Qingmeng Li et al.

对用户自定义的多个衍射峰区间进行寻峰扣背底，生成3D分布图

打开 →

O

Overleaf 助手

by Fengyao Hou

Hai Overleaf 平台的智能写作助手，提升LaTeX文档编辑效率

打开 →

DrSai_BESIII_v2.0.0

有什么我能帮您的吗?

+

建议

Psip -> pi+ pi- [J/psi -> Lambda Lambdabar]

psi(4260) -> K+ K- [J/psi -> e+ e-]

psi(3770) -> pp

J/psi -> mu+ mu-

Psip -> pi+ pi- [J/psi -> p pbar eta]

psi(4260) -> pi+ pi- [J/psi -> mu+ mu-]

PDF 文件上传与解析

支持批量上传、智能解析、实时预览

开发者指南 文件管理

解析模式

A PDF parser for MinerU v1.0



拖拽 PDF 到此，或 点击选择文件

支持单个 ≤ 20MB，PDF 格式

文件列表 (0)

开始解析

实时解析结果

全部结果 解析中 已完成



暂无实时解析结果

上传 PDF 开始解析，或前往文件管理查看历史结果

文件管理

支持与服务



- 联系方式
- 高能所计算中心
- 邮箱：
 - hepai@ihep.ac.cn
- 反馈帮助：
 - helpdesk.ihep.ac.cn
 - zhangyiyu@ihep.ac.cn
- 微信群
- 技术支持文档: <https://ai.ihep.ac.cn/hai-docs>



群聊: HAI集群用户群 (HepAI)



该二维码7天内(3月31日前)有效, 重新进入将更新

总结



- HepAI平台秉持数据、算法、算力、应用“四位一体”的核心理念，致力于打造推动AI赋能科学研究的共性基础设施。
- 未来计划
 - 扩充AI算力，以满足更多研究人员的AI开发工作。
 - 完善AI平台的模型和数据集服务，使用户更好的迭代和管理自己的模型和数据集。
 - 完善AI开发 workflow，打通模型-数据-算力瓶颈，提供快捷训练和自定义训练服务。

我们诚邀各位用户登录ai.ihep.ac.cn使用AI算力服务，开发自己的AI模型，提升您的研究和项目。
我们欢迎各位用户与我们开展深度的AI研究合作



谢谢



群聊：HAI集群用户群（HepAI）



该二维码7天内(3月31日前)有效，重新进入将更新