中国科学技术大学
University of Science and Technology of China

基于容器的计算实训平台

-中国科大计算实训平台建设

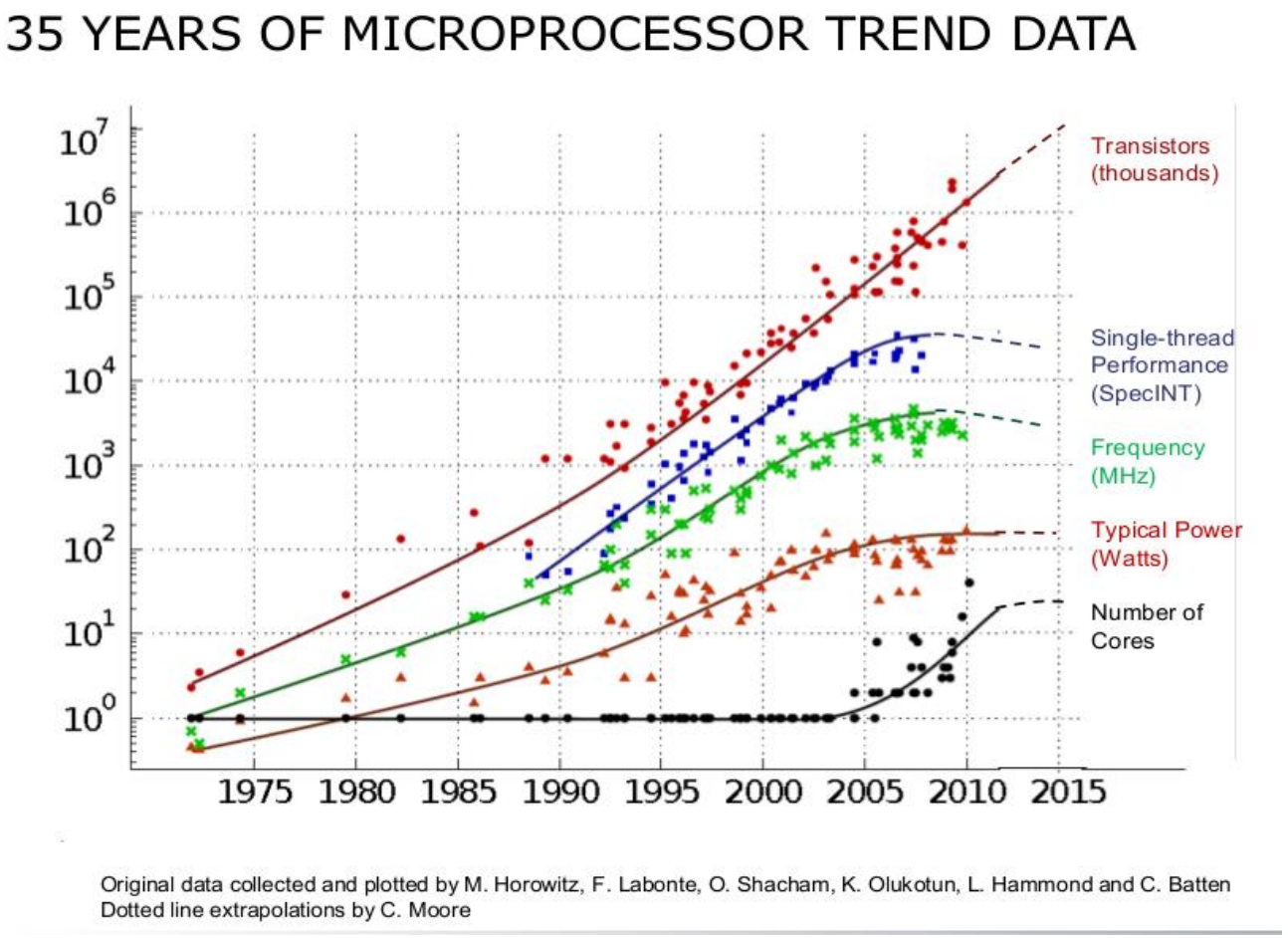
李会民

http://scc.ustc.edu.cn

2019-11-14

- 为什么要建？
- 系统架构及功能
- 运行数据
- 未来展望

功耗墙和并行计算



功耗问题 → 多核技术

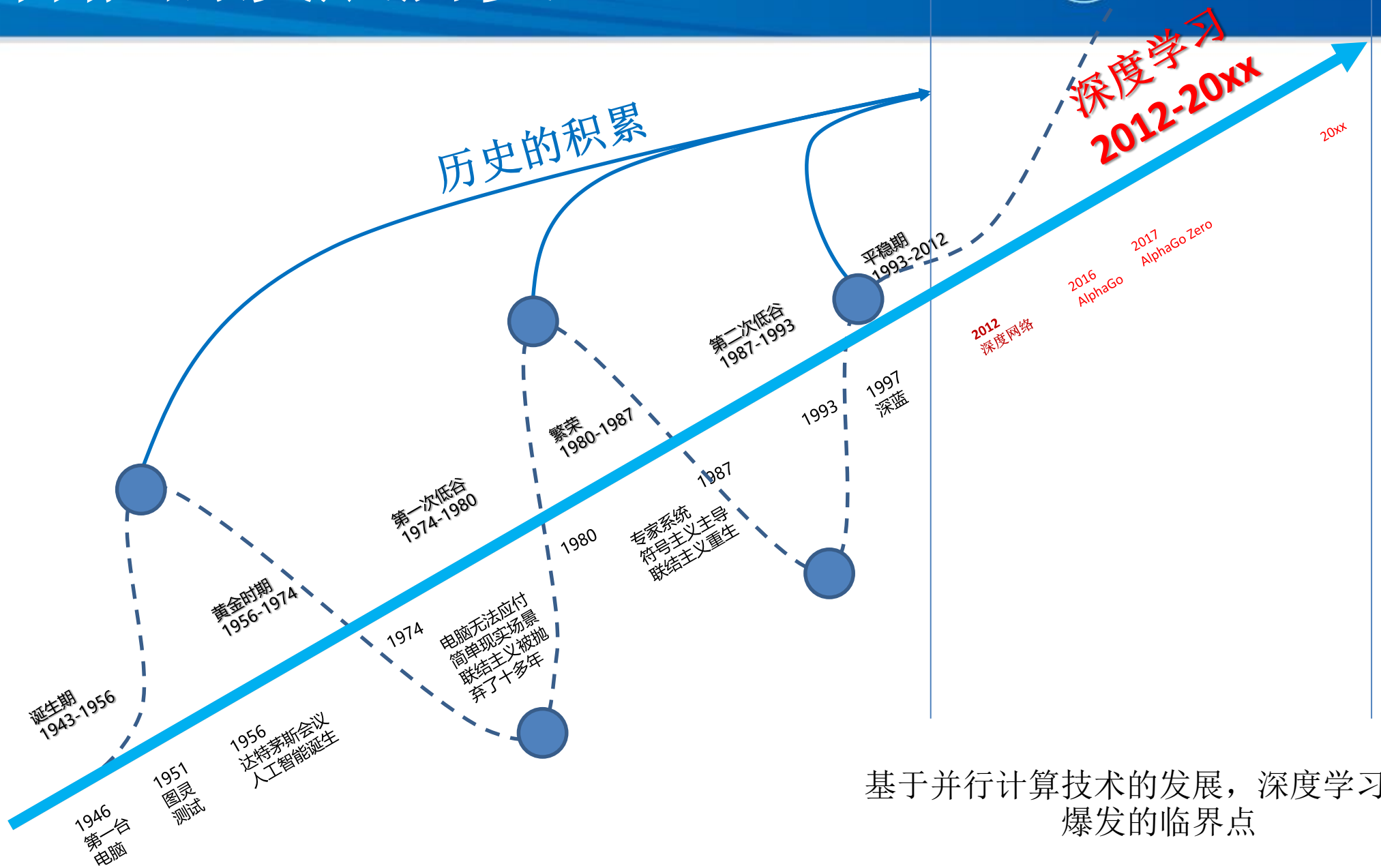
+

多核技术 → 依靠主频加速应用的“免费午餐”时代结束

→

并行化及并行化优化显得尤为重要

人工智能的发展历史



基于并行计算技术的发展，深度学习处于爆发的临界点

HPC与AI的融合

中国科学技术大学
University of Science and Technology of China

HPC应用服务平台

- 面向用户提供可视化科研实验应用服务
- 面向用户提供数据管理服务

HPC实训和实验环境

- 提供体系化的基础知识
- 为新手提供实验环境，避免与科研实验环境冲突

HPC运行环境和工具

- 编译环境、并行环境、数学库
- 调度系统

HPC基础云平台

- 基于物理机实现自动化资源供给
- 平台使用情况运维和管理

HPC硬件平台

- 计算服务：CPU/GPU
- 存储服务：磁盘阵列/文件系统

AI应用服务平台

- 面向用户提供可视化科研实验应用服务
- 面向用户提供数据管理服务

AI实训和实验环境

- 提供体系化的基础知识
- 为新手提供实验环境，避免与科研实验环境冲突

深度学习框架

- Tensorflow、Caffe等深度学习框架和相关工具库

AI运行环境和工具

- 编译环境、并行环境、数学库
- 调度系统

AI基础云平台

- 基于物理机或容器实现自动化资源供给
- 平台使用情况运维和管理

AI硬件平台

- 计算服务：CPU/GPU
- 存储服务：磁盘阵列/文件系统

中国科学技术大学简介

中国科学技术大学
University of Science and Technology of China

隶属于中国科学院

首批世界一流大学和一流学科（双一流）建设高校

首批“985工程”C9高校之一
首批“211工程”高校之一

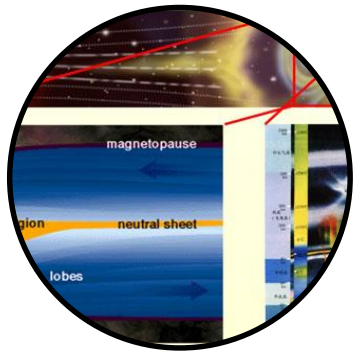
唯一参与国家知识创新工程的高校

国家实验室或研究中心：2个

其他国家级科研机构：10个

国家重点科技基础设施：4个

院省部级重点科研机构：55个



国家理科基础科学研究和教学人才培养基地：6个

国家生命科学与技术人才培养基地：1个

一级学科国家重点学科：8个

二级学科国家重点学科：4个

国家重点培育学科：2个

安徽省一级重点学科：18个

中国科学院和中国工程院院士：55人

发展中国家科学院院士：17人

国家万人计划领军人才：30人
青年拔尖人才：13人
国家杰出青年科学基金获得者：115人
优秀青年科学基金获得者：108人
教育部长江学者（含青年）：51人
国家级教学名师：7人

以前沿科学和高新技术为主的综合性全国重点大学

中国科大超级计算中心简介

中国科学技术大学
University of Science and Technology of China

2002年底，学校决定建设第五公共服务平台，即科学计算平台（超级计算平台），2003年10月初步建成

学校六大公共实验中心之一，挂靠网络信息中心

本着推动科学计算、促进人才培养的建设方针

- 面向校内所有有高性能计算需求的科研院系、实验室、教师、学生提供高性能计算服务
- 建有“研究生超级计算实验训练中心”培训超级计算相关人才
- 建有苏州分中心提供基于寒武纪智能芯片的超级计算资源等

积极对外合作

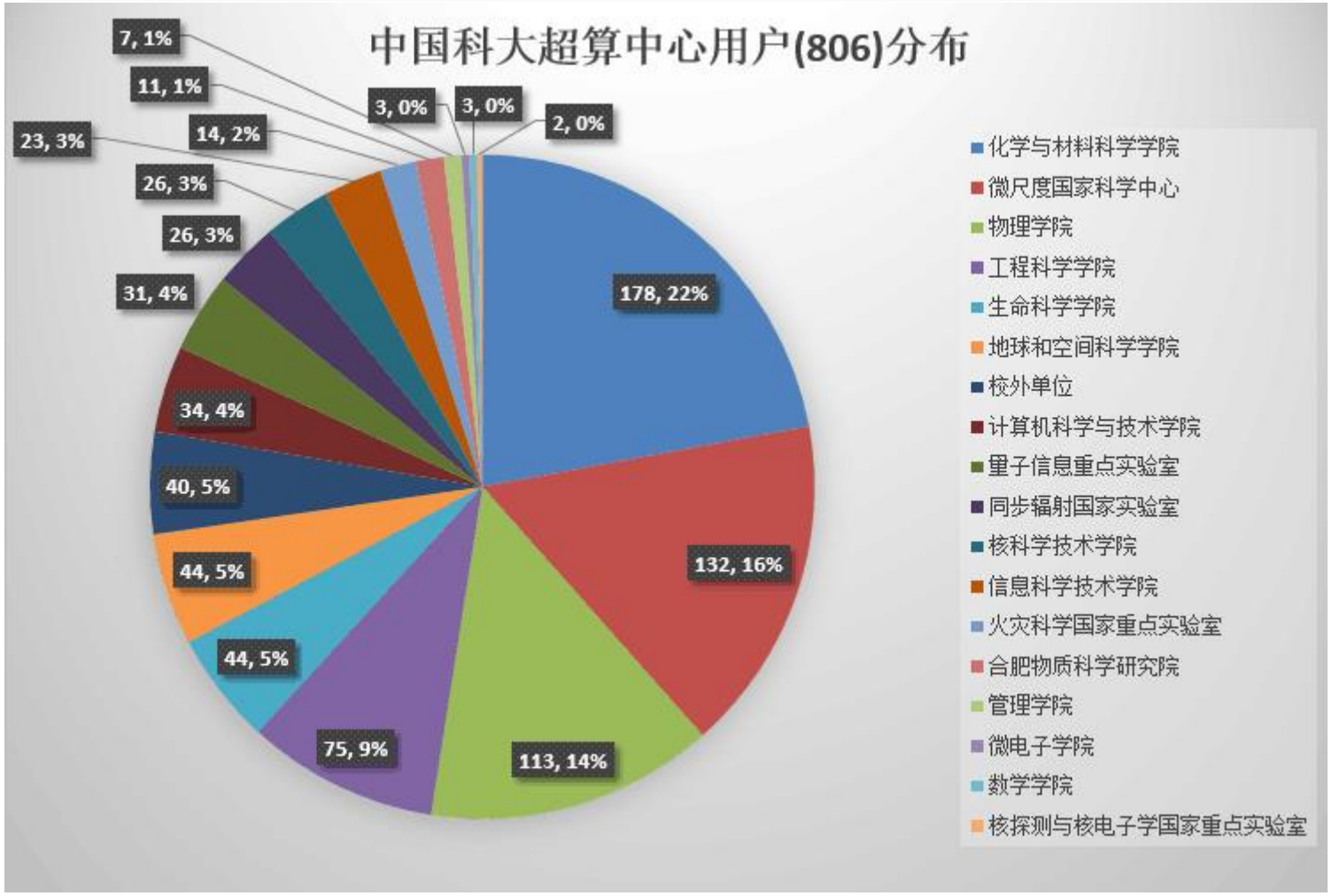
- 中国国家网格（CNGrid）合肥中心（2005年）和中国国家网格(CNGrid)合肥运行中心（2016年）
- 中国科学院超级计算环境和GPU分中心（2010年）和合肥中心（中科大）（2014年）
- 国家超级计算天津中心中科大分中心（2012年）
- 中国教育与科研网格项目（ChinaGrid）中科大子节点（2013年）
- 超级计算创新联盟理事单位（2014年）
- 安徽省高校科研协作高性能计算公共平台（2014年）：对安徽省教育和科研计算机网高校用户提供高性能计算服务
- 与合肥微尺度物质科学国家研究中心材料基因团队共建高通量量子材料基因库计算平台（2016年）

超算系统

- 运行中：TC4600百万亿次超级计算系统：2600万元，508个计算节点，总理论计算能力648.53万亿次/秒，其中CPU：487.49万亿次/秒，GPU：139.76万亿次/秒、MIC：21.28万亿次/秒
- 建设中：支持海量数据处理的高IO性能计算模拟系统，5000万元，国内高校最大单套系统，748个计算节点：29520颗Intel Xeon Scale 6248 CPU核心、 20块NVIDIA Tesla V100 GPU卡、 960颗华为海思Hi1620 CPU核心，理论峰值：CPU：2.3PFlops，GPU：0.14PFlops

超算用户

覆盖所有学院和主要实验室的教师和学生，
目前有806个系统账户（含校外用户40个）



- 国家同步辐射实验室
- 合肥微尺度物质科学国家研究中心
- 量子信息与量子科技前沿协同创新中心
- 类脑智能技术及应用国家工程实验室
- 稳态强磁场科学中心
- 高能量密度物理联合实验室
- 国家高性能计算中心（合肥）
- 核电子学国家重点实验室（联合）
- 热安全技术国家地方联合工程研究中心
- 大尺度火灾国际联合研究中心
- 火灾科学国家重点实验室
- 中国科学院合肥大科学中心
- 中国科学院量子信息与量子科技创新研究院
- 中国科学院星系宇宙学重点实验室（联合）

用户成果产出与积分奖励

中国科学技术大学 University of Science and Technology of China

超算中心支持用户取得了丰硕的成果（不完全统计）

自2004年到2018年，支持发表致谢论文1171篇，其中Nature及其子刊、PRL与JACS三类顶级期刊59篇，2019年已支持发表2篇Science/Nature正刊论文、3篇子刊论文

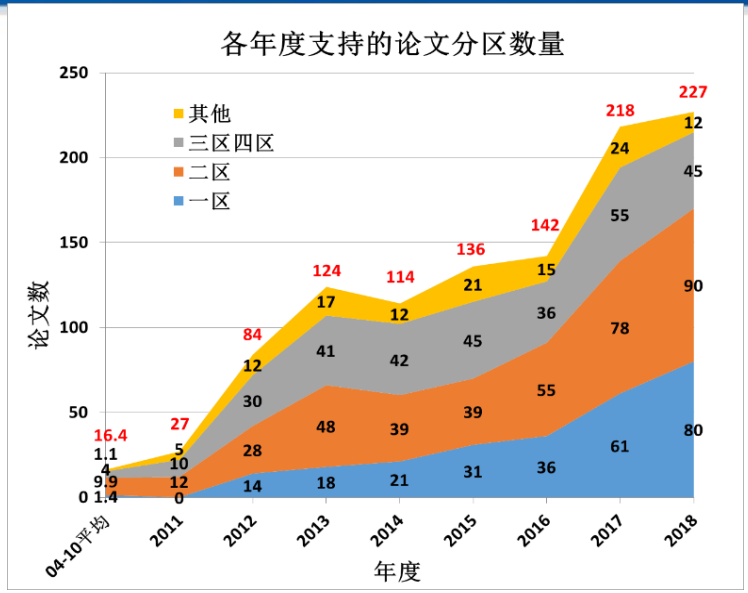
- 支持的高质量论文数量多，且分布广泛，支持的约20篇Nature/Science系列论文来自十多个课题组
- 发表论文的计算领域主要集中在原子分子材料计算领域，包括2018与2019年的全部顶级论文

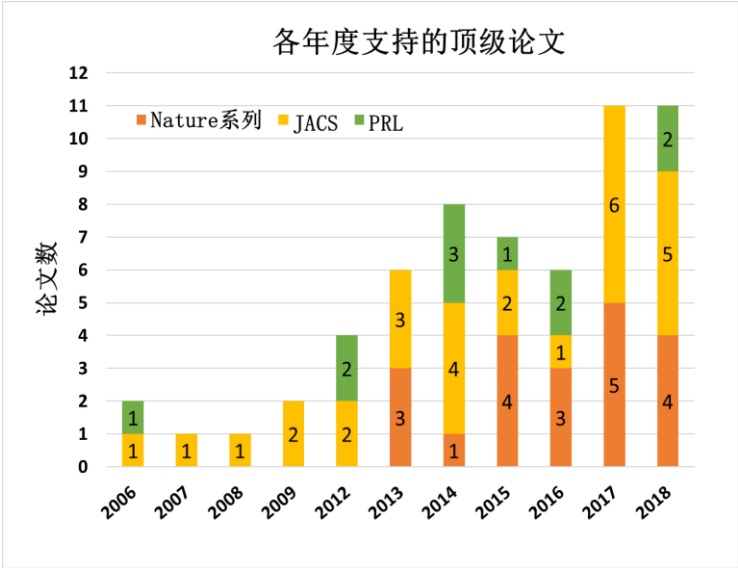
2018年支持了96项科研项目，总经费1.76亿元，产出227篇论文，一区、二区文章占比分别为35%、40%

2018年奖励1491积分（1积分奖励免费机时1万CPU核时）

- 化学与材料学院891积分，60%
- 物理学院297积分，20%
- 合肥微尺度物质科学国家研究中心88分，6%
- 物理化学类占比86%。

2019年，超过40%校新闻首页报道的重大科研成果由中心支持





论文目录：http://scc.ustc.edu.cn/zclw/

用户缺乏Linux等基础

中国科学技术大学
University of Science and Technology of China

超算用户Linux基础有待提高

- 怎么建立目录
- 怎么查看文件内容
- 内存不够，能否扩200G内存

编译环境不熟悉

- 怎么编译
- 编译出错怎么办

作业调度系统不熟悉

- 如何提交作业
- 如何查看作业

计算机教学

- C、算法等教学

专业软件教学

- Materials Studio

技术支持与培训服务

封装编译运行环境、用户应用等，方便用户使用

详尽的文档

- 用户手册：17种；技术文章：19种；常见问题

用户培训

- 研究生超级计算实验训练中心
- 培训讲座：自己主办（17种），邀请校外专家等

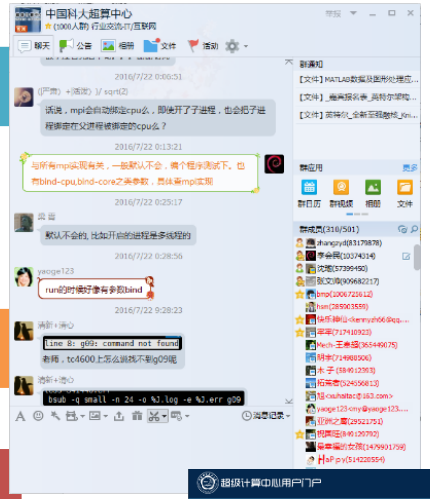
邮件、电话、QQ即时通讯

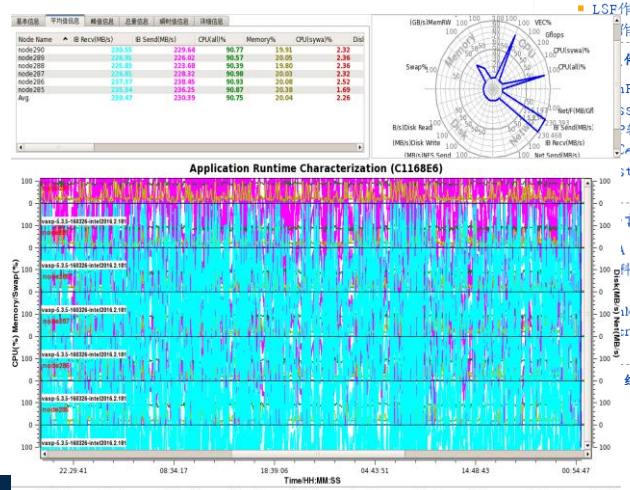
- QQ用户群：面向普通用户，及时提供技术支持，969人
- QQ运维群：面向用户自建超算的维护人员，帮其提高运维水平，45人，来自20多个实验室
- QQ托管群：为托管用户服务

主动服务

- 监控用户作业，发现问题，主动处理

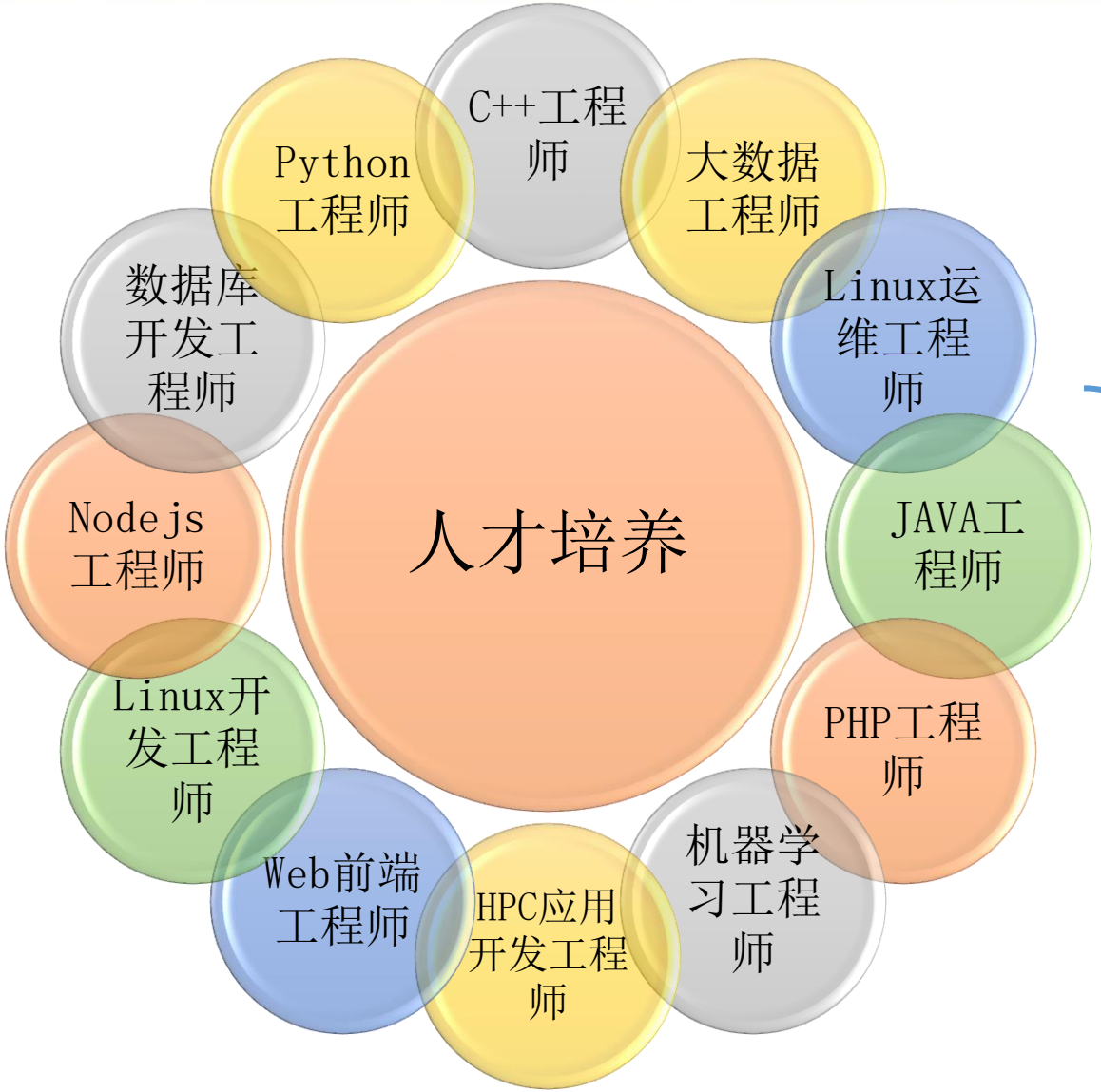
与外力合作提高运维及用户服务技术水平





- 常见使用问题
  - 常见使用问题
- 培训讲座
  - 2016年秋学期用户培训资料
  - 2016年6月Intel高性能计算培训
  - 2015年11月用户培训资料
  - 2015 MATLAB中文论坛现场交流会-合肥站
  - 2014年9月国家超算天津中心培训交流会
- 曙光TC4600百万亿次超级计算系统
  - 曙光TC4600百万亿次超级计算系统用户使用指南
  - 曙光TC4600百万亿次超级计算系统用户使用指南-简略
  - Intel 2017.0.098编译器、调试器、MKL及MPI资料
  - Intel 2016.0.109编译器、调试器、MKL及MPI资料
  - Intel 2015.1.133编译器、调试器、MKL及MPI资料
- ChinaGrid高性能计算集群
  - ChinaGrid高性能计算集群使用指南
  - ChinaGrid高性能计算集群MIC服务器使用指南
  - Intel 2013版编译器、调试器、MKL及MPI资料
- 刀片及胖节点超级计算系统
  - 刀片及胖节点超级计算系统用户使用指南
  - Intel 2011版编译器、MKL、调试器资料
- LSF作业调度系统
  - LSF作业调度系统(8.3)
  - 作业调度系统(6.0)

人才需求



- Linux基础
- Linux服务
- VIM编辑
- VPN管理
- Git管理
- Shell基础
- 网络基础
- Python基础
- MySQL基础
- Linux防火墙
- MongoDB基础
- Linux系统监控
- Redis基础
- ELK分布式分析系统
- SDN基础
- Linux Web运维

- 为什么要建？
- 系统架构及功能
- 运行数据
- 未来展望

实训实验平台的建设

中国科学技术大学
University of Science and Technology of China

基础设施

选择具备有一定计算能力的高密度基础设施，在满足AI/HPC实训需求的同时，考虑教师授课需求，支持计算相关方向的教学

高密度的基础设施

实训实验平台

基于云平台的实验环境，支持学生的学习和实验，按需自动调度的全自动化管理智能平台。

灵活的实训平台

学习内容

围绕学生阶段特点，建立多种内容的知识体系、学习路径，提供有针对性、理论实践相结合的学习内容。

定制化的学习内容

系统架构

中国科学技术大学 University of Science and Technology of China

交互层

- 学员门户
- 教师门户
- 管理员门户

内容层

典型实训内容

- 图像识别
- 语音识别
- Python基础
- Linux基础
- 编译环境
- 调度系统
- C/C++基础
- 量化计算
- 生物计算
- 大数据处理
- 数据标注
- 系统优化
- 分布式数据库
- 系统运维

业务层

平台业务能力

- 自助实训
- 实训辅助
- 自主试验
- 实训环境预约
- 用户管理
- 平台数据分析
- 自动连接管理
- 智能资源调度
- 资源审批
- 业务扩容
- 实例管控
- 实训课程开发

工具层

工具

- TensorFlow
- Caffe
- Numpy
- OpenCV
- OpenMP
- Intel MKL
- Intel MKL

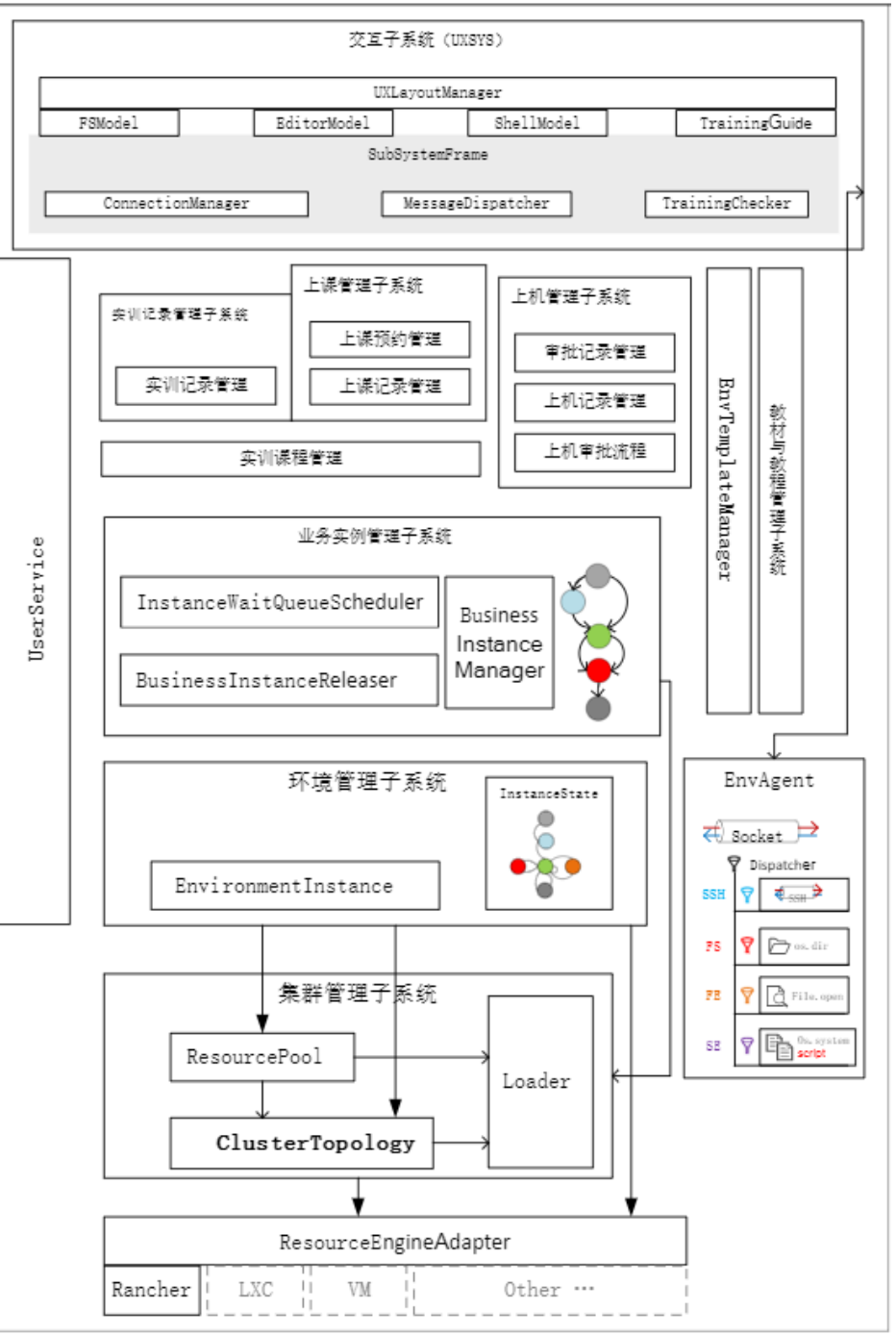
数据

- 表情识别训练集
- 手写数字训练集
- 诗歌文本训练集
- 语音数据训练集
- HPC行业应用数据
- 并行优化数据

平台层

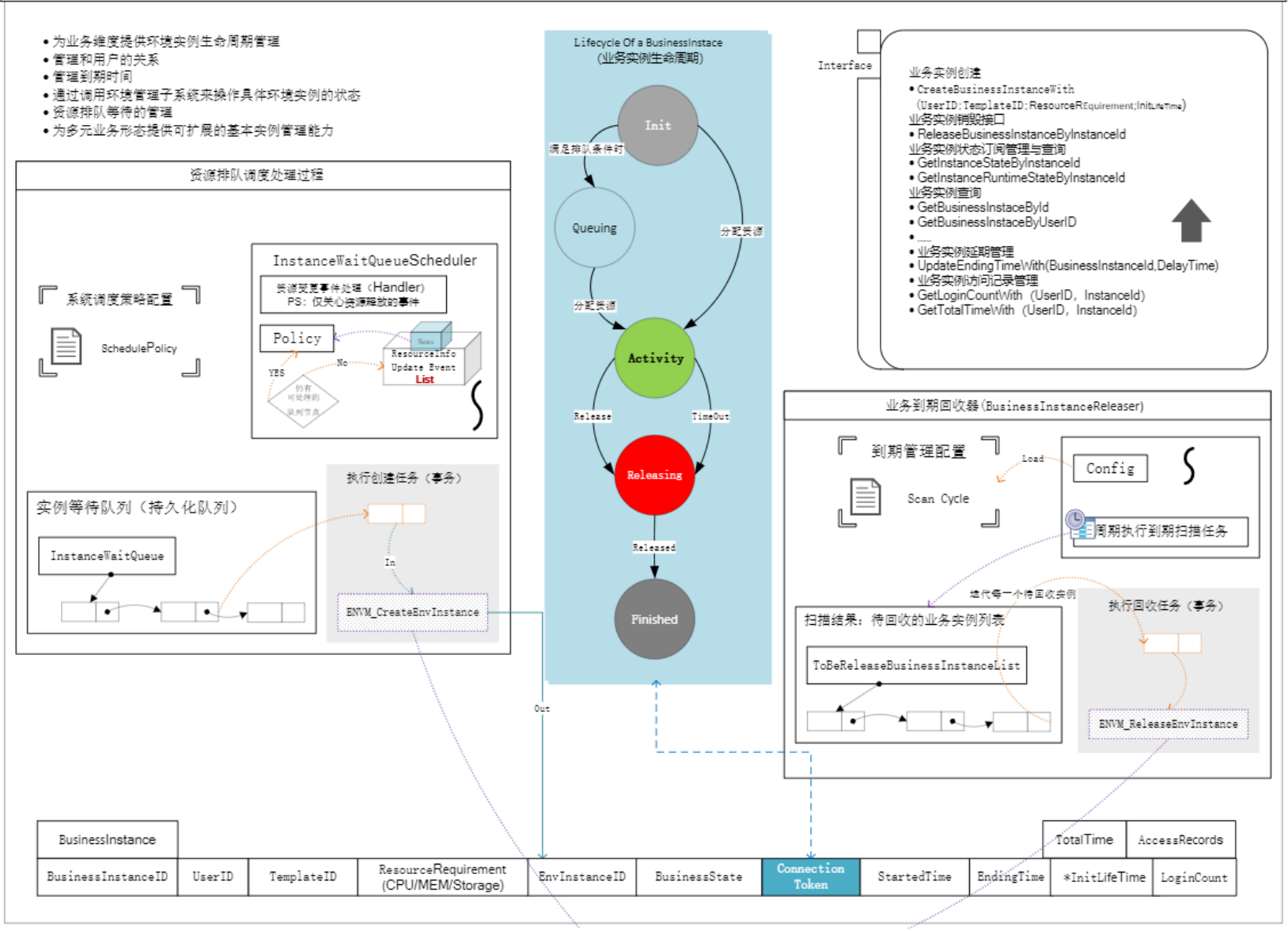
基础云平台（支持灵活的资源分配）

实训平台架构全景图 （Level 0层）

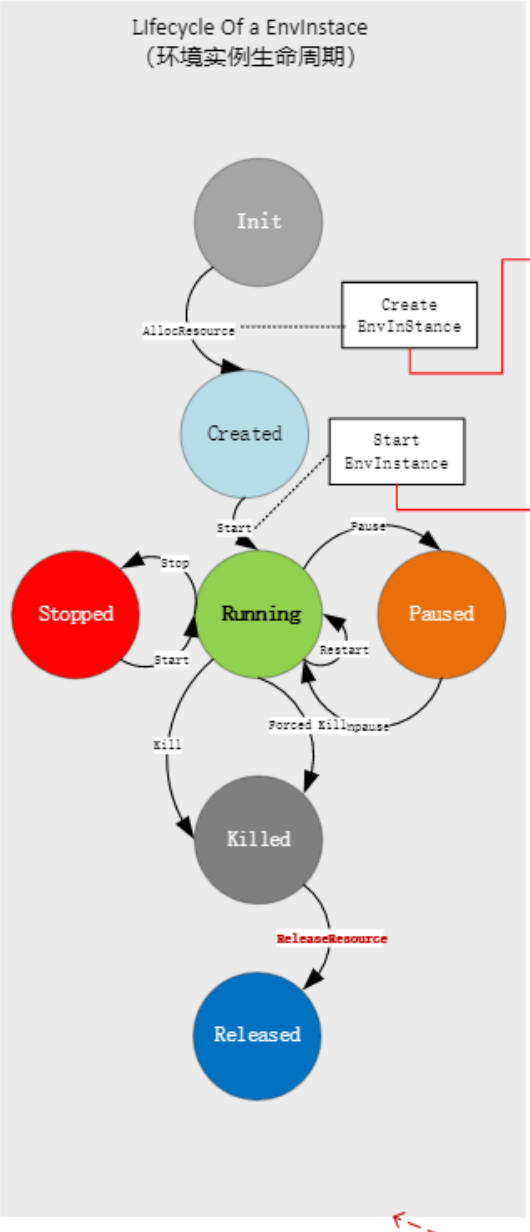


业务实例管理子系统（BusinessInstanceManagement）【BIM】

- 为业务维度提供环境实例生命周期管理
- 管理和用户的关系
- 管理到期时间
- 通过调用环境管理子系统来操作具体环境实例的状态
- 资源排队等待的管理
- 为多元业务形态提供可扩展的基本实例管理能力



环境管理子系统（ENVM）



环境管理子系统主要是资源调度申请释放、环境实例创建销毁等生命周期管理，这一层是面向硬件和环境的管理层，与具体用户和业务无关，负责与资源池管理和资源引擎进行交互，聚焦于环境实例的生命周期管理。具体职责如下：

1 管理环境实例 创建、销毁、暂停等状态，以及在环境实例创建和销毁时具体资源的申请与释放

1.1环境实例的健康状态检查与更新

1.2环境实例的相关信息查询

2 管理具体环境的访问控制信息（IP地址、用户名、密码等）

两段式实例创建过程，创建实例只是完成资源分配，启动实例才会运行真正的Docker

Interface

环境实例创建与运行

- CreateEnvInstanceWithTempIdAndCustomeInfoAndResourceDescription
- StartEnvInstanceWithEnvId

环境实例销毁接口

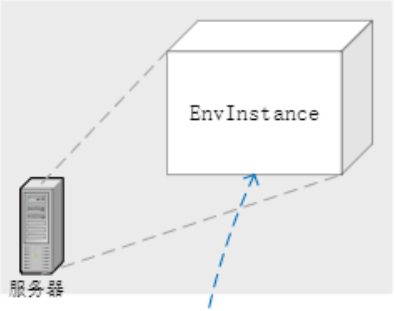
- DestroyEnvInstanceByInstanceId
- ForcedDestructionEnvInstanceByInstanceId

环境实例状态订阅管理与查询

- RegisterEnvInstaceStateChangeHandlerWithInstaceId
- GetStateByInstanceId

环境实例查询

- GetEnvInstaceById
- GetEnvInstaceInfoById



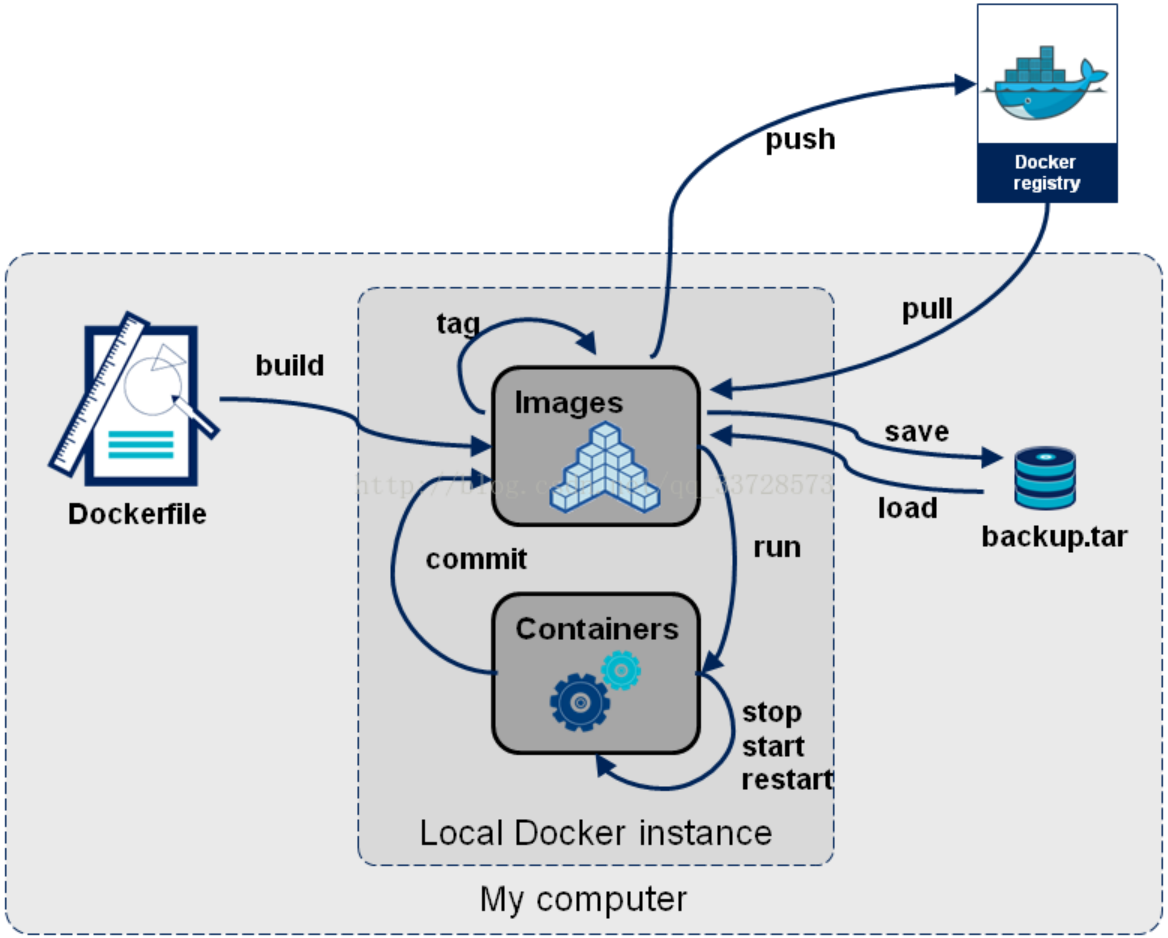
Write to FS

核心对象：环境实例

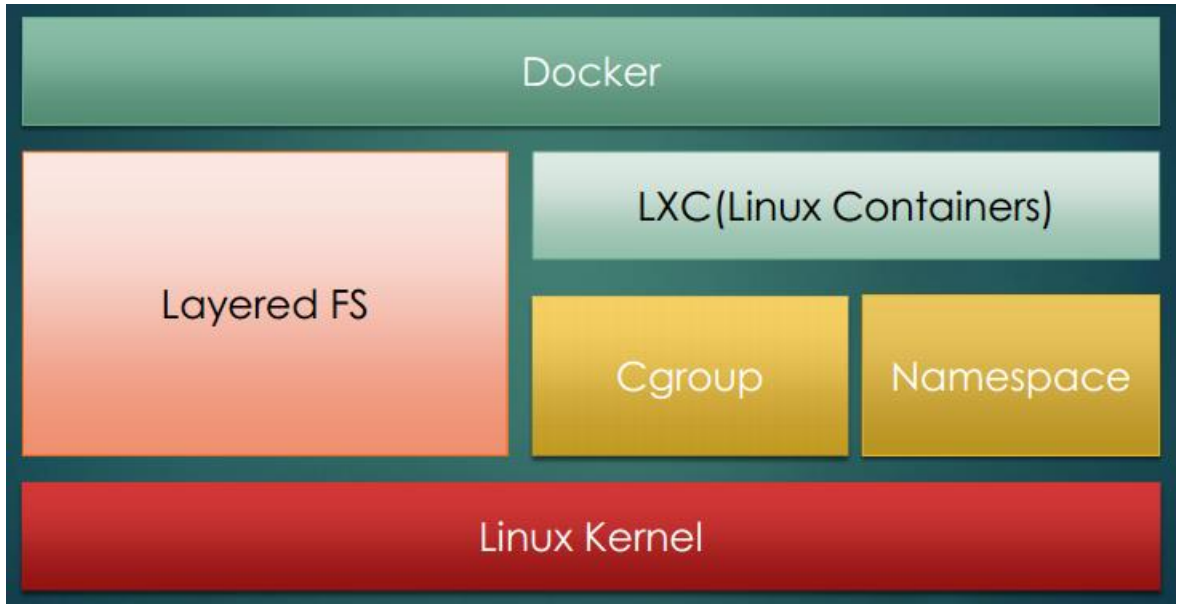
EnvironmentInstance

EnvInstanceId	ByTemplateId	InstanceInfo (IP/LogoinName/PWS)	NodeId	InstanceState	AllocationRecordID	CustomInfo

容器工作流程和架构



工作流程



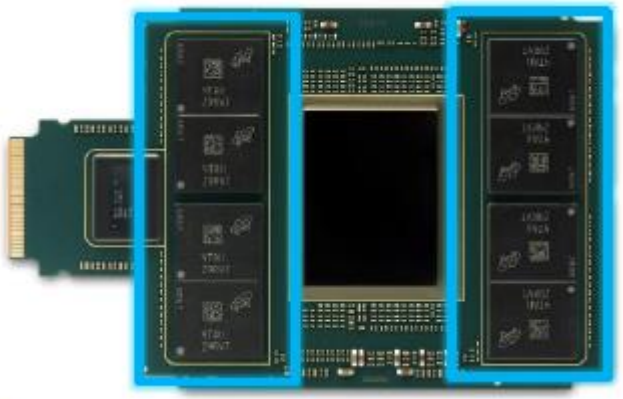
架构

编排工具

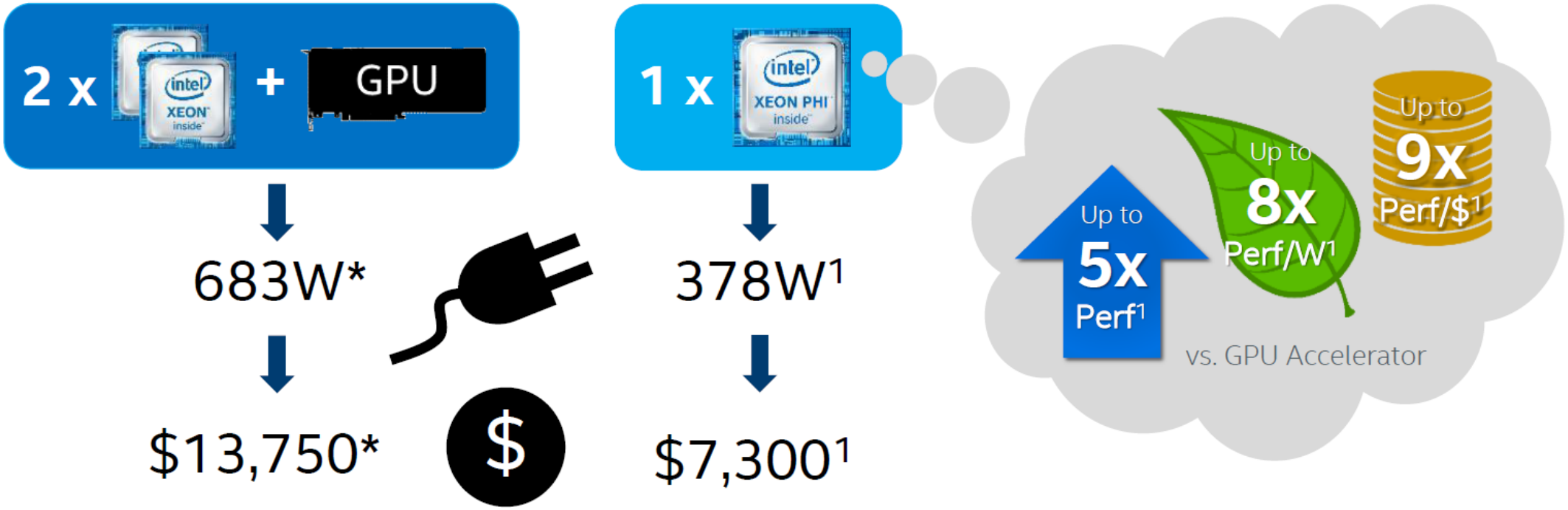
比较项目	Kubernetes	Swarm Mode	Mesos+Marathon
来源	Google	Docker原生	Apache mesos
开发语言	Go	Go	C++/Scala
生产经验	★	暂无	★★
学习成本	★★★	★	★★
运维成本	★★	★	★★★
社区活跃度	★★★	★	★
适用场景	对外服务，或者需要提供高可靠服务的场景	对内使用，当作私有云来使用，或者对容器的可靠性要求不高	大型生产环境，团队有足够的Linux分布式经验

硬件支撑环境

采用高密度，众核平台



芯片内置内存



软件平台

- 系统资源调度
- 系统环境配置
- GPU环境配置

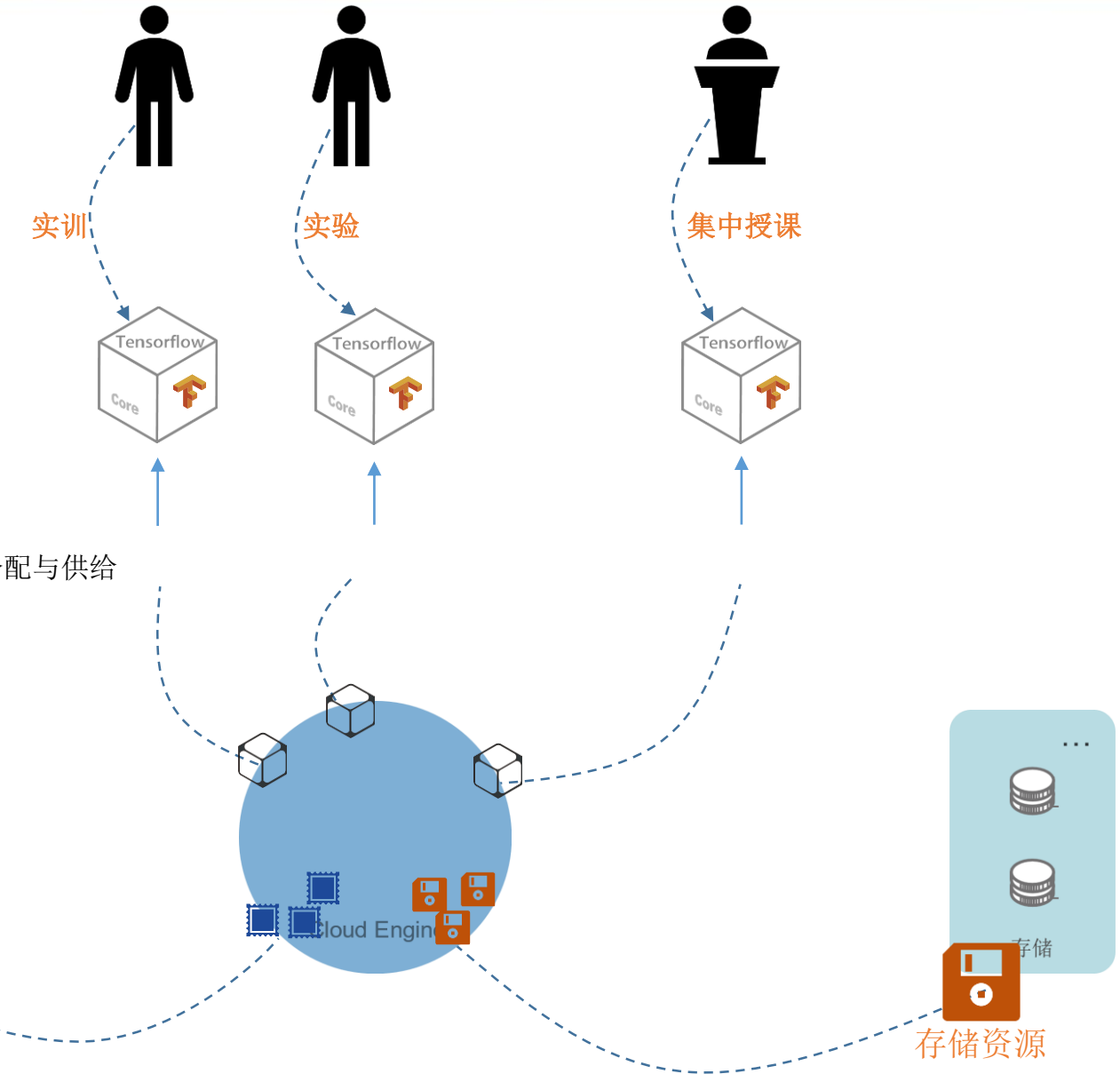
环境免设置，免配置

- 软件环境配置
- 深度学习代码配置
- 运行代码框架

软件实训环境，免配置

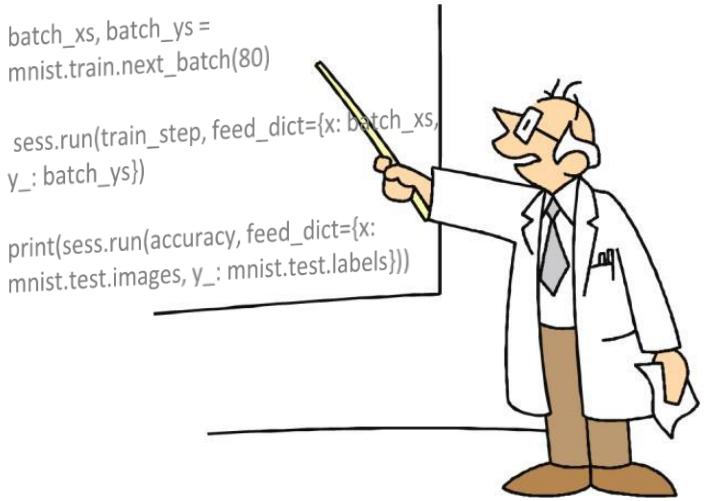
- 训练集&测试集

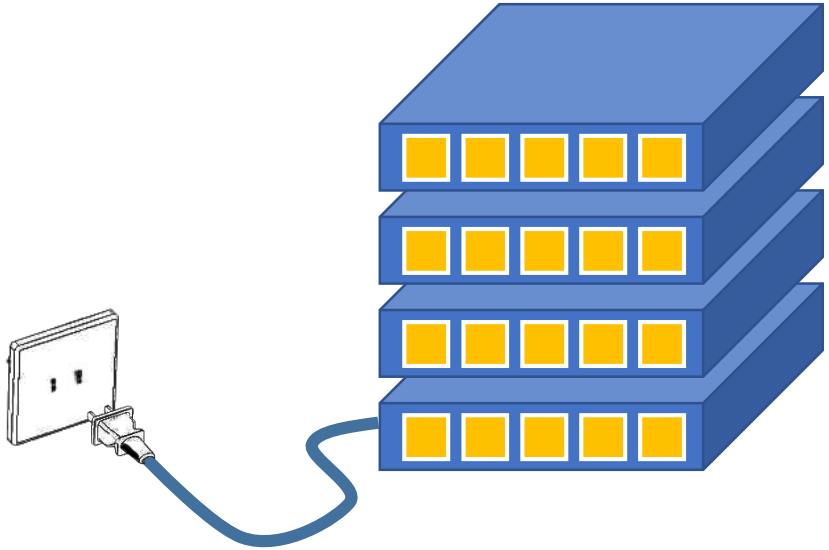
内置实训数据



CentOS 7 + Docker

软件平台





- 自动申请、自动分配、免配置、聚焦于内容学习
- 实例练习，从理论到代码融会贯通
- 教学、实训一体化：边学边练
- 分时共享、自动回收、免管理

HPC基础和实训内容

中国科学技术大学
University of Science and Technology of China

Linux基础

- Linux简介
- 系统运行
- 文件和目录
- 进程
- 工具
- Shell
- VI/Emacs
- 推荐软件

作业调度系统

- LSF简介
- 超算中心配置
- LSF使用
- PBS简介
- PBS使用
- Slurm简介
- Slurm使用

编译器

- Intel C/C++
- Intel Fortran
- PGI C/C++
- PGI Fortran
- GNU C/C++
- GNU Fortran

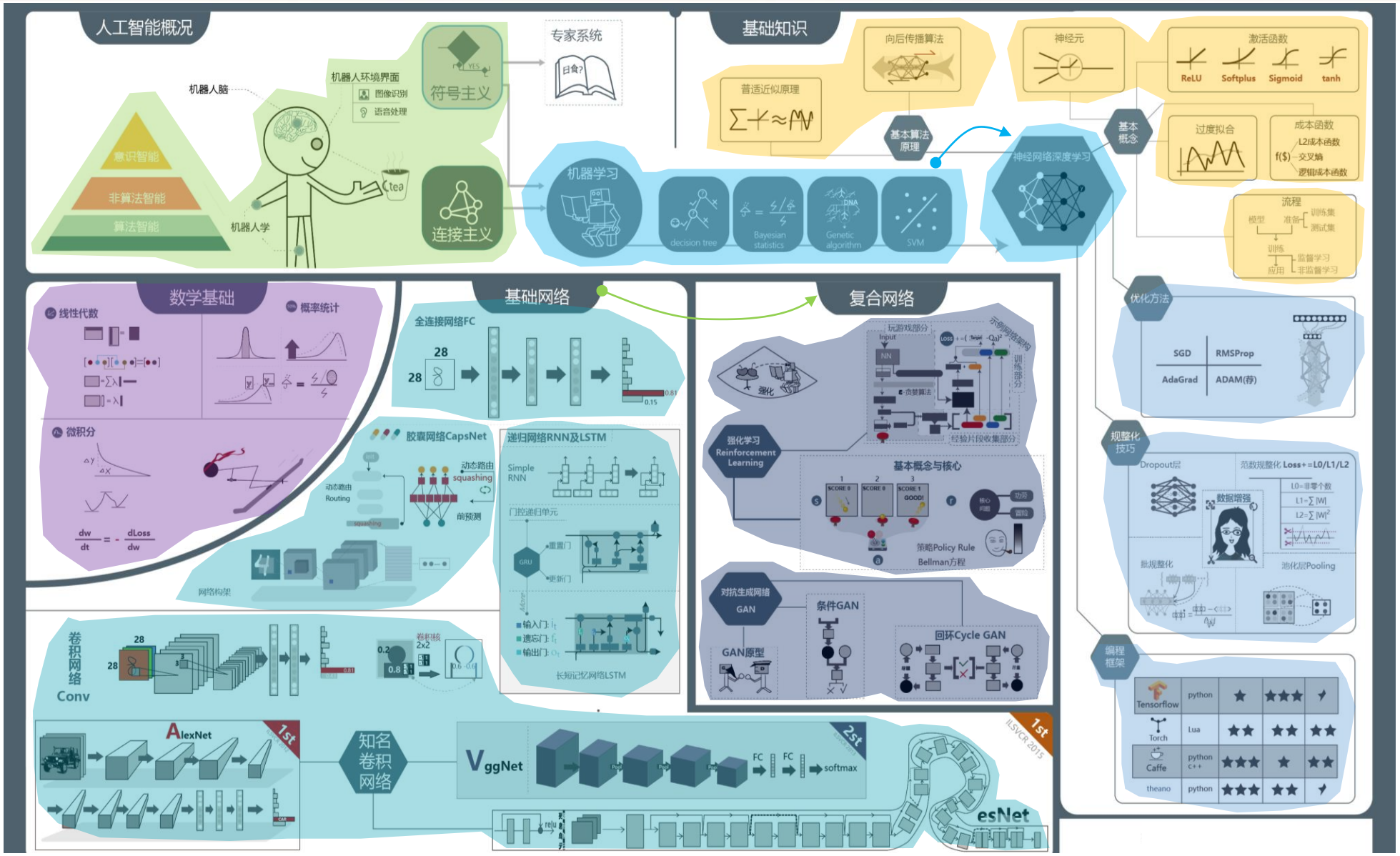
数学库

- Intel MKL
- ACML
- IMSL
- NetLib
- GotoBLAS
- ATLAS

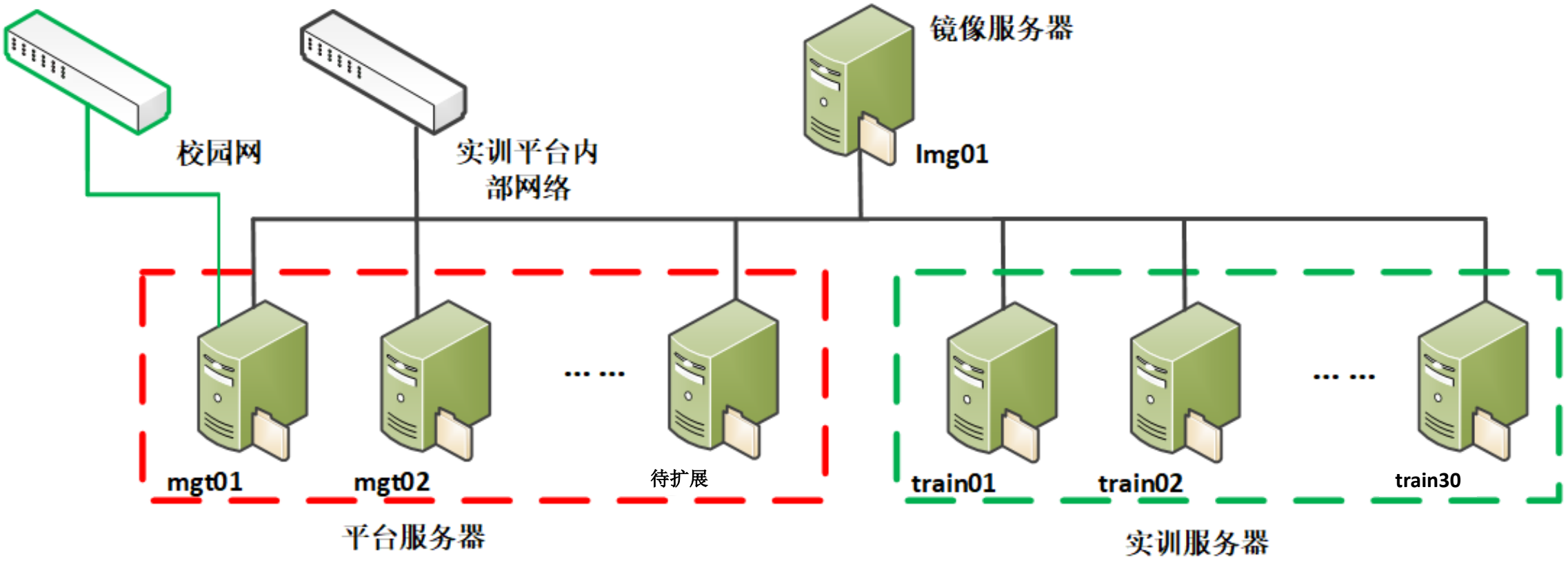
并行环境

- Intel MPI
- Open MPI
- 应用性能优化

AI基础和实训内容


人工智能概况
机器人脑
意识智能
非算法智能
算法智能
机器人学
机器人环境界面
图像识别
语音处理
符号主义
专家系统
连接主义
机器学习
decision tree
Bayesian statistics
Genetic algorithm
SVM
基础知识
向后传播算法
普适近似原理
基本算法原理
神经元
激活函数
ReLU
Softplus
Sigmoid
tanh
基本概念
过度拟合
成本函数
L2成本函数
交叉熵
逻辑成本函数
神经网络深度学习
流程
模型
准备
训练集
测试集
训练
应用
监督学习
非监督学习
数学基础
线性代数
概率统计
微积分
基础网络
全连接网络FC
胶囊网络CapsNet
动态路由
squashing
前预测
网络构架
递归网络RNN及LSTM
Simple RNN
门控递归单元
GRU
重置门
更新门
输入门
遗忘门
输出门
长短记忆网络LSTM
复合网络
强化
强化学习 Reinforcement Learning
基本概念与核心
策略Policy Rule
Bellman方程
对抗生成网络 GAN
GAN原型
条件GAN
回环Cycle GAN
卷积网络 Conv
卷积核
AlexNet
1st
知名卷积网络
VggNet
2st
FC
softmax
esNet
ILSVCR 2015
优化方法
SGD
RMSProp
AdaGrad
ADAM(荐)
规整化技巧
Dropout层
范数规整化 Loss+=L0/L1/L2
数据增强
批规整化
池化层Pooling
编程框架
Tensorflow
python
Torch
Lua
Caffe
theano

系统部署情况



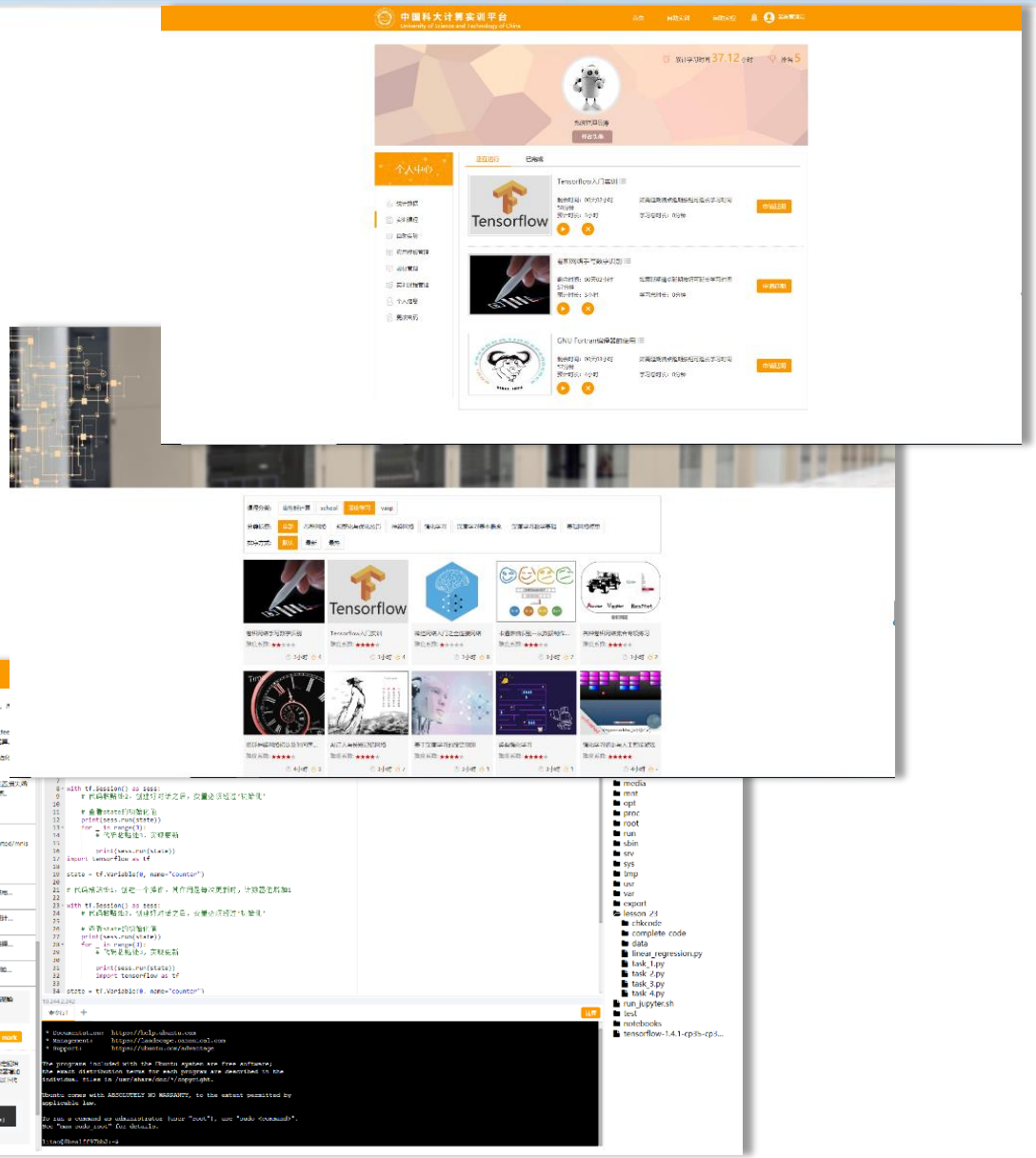
整体包含2040个物理CPU核

域名

- http://training.ustc.edu.cn
- http://training.ah.edu.cn

学员视角

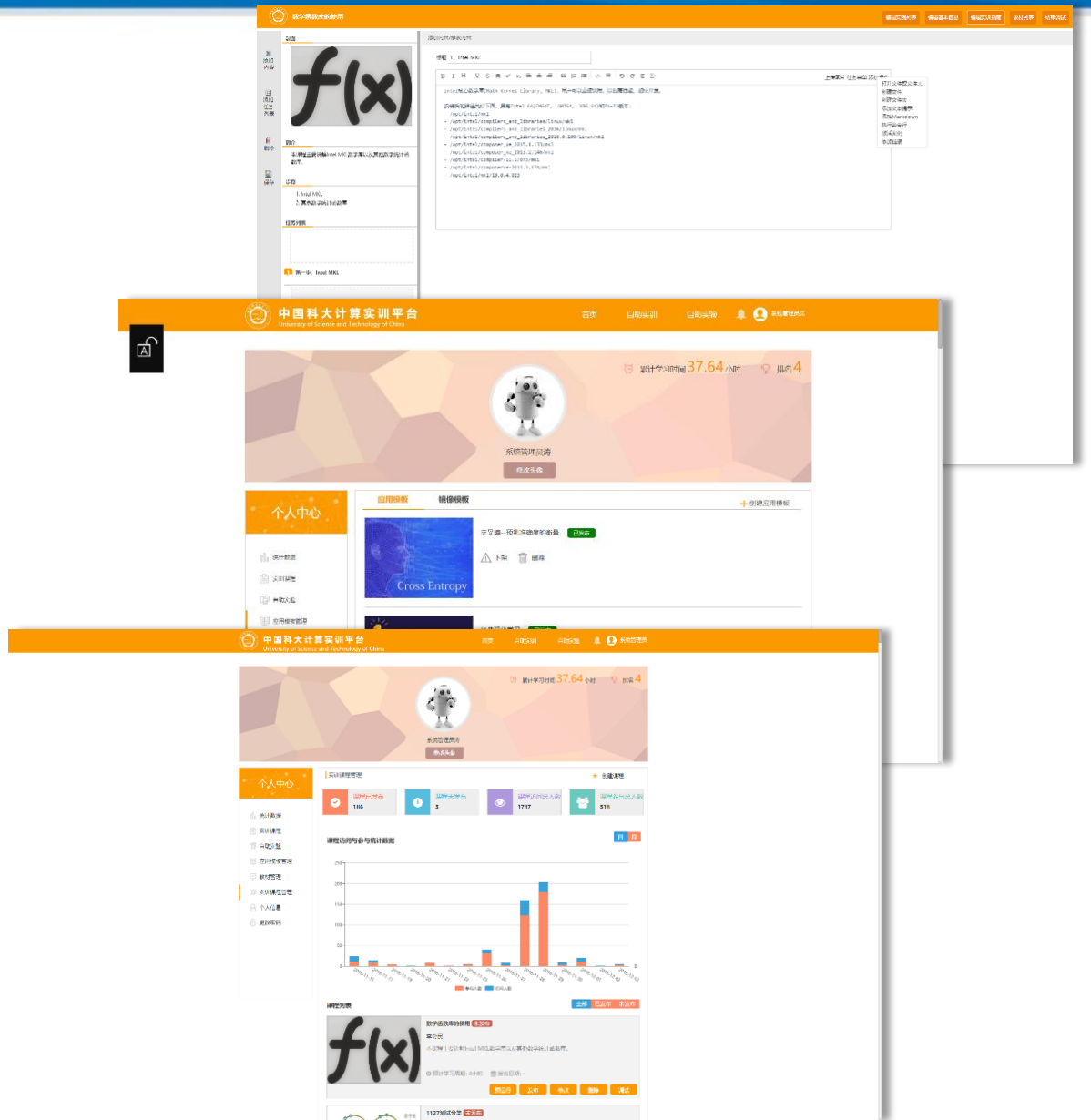
中国科学技术大学
University of Science and Technology of China



功能	价值
查看、参加、管理自助实训	随老师的课程讲解，巩固基础知识，完成理论练习，根据需求主动或被动释放环境，以节省系统资源
学员技能能力分析图谱	根据实训的情况，推测学员擅长的技能和薄弱的环节
实训指南系统	进行理论提醒与辅导，代码提示，并能够设定教学中待学员解答的问题，帮助学员消化理解理论知识，同时帮助学员更便捷的使用系统环境
按需申请实验环境	在完成理论学习后可以按照自己创意尝试应用理论知识，系统提供友善的界面和环境
可申请并使用集群级环境和GPU环境	让学员可以完成大型集群的实训课程与试验以及GPU环境与试验
参加特定课程	参加一次预定好的课程，避免上课时没有环境
环境操作界面	可以编辑保存代码，从环境中下载文件或上传文件，支持多文件编辑。 命令行支持复制粘贴 环境状态查看……
完整的深度学习实训课程	系统为学员提供完整的深度学习知识体系所必须的实训课程，这些课程简化了程序的结构和实现的复杂度，主要是让学员完成理论和技术的学习。
深度学习知识地图	学员可以浏览地图从宏观上理解自身技能的欠缺和下一步要学习的知识点

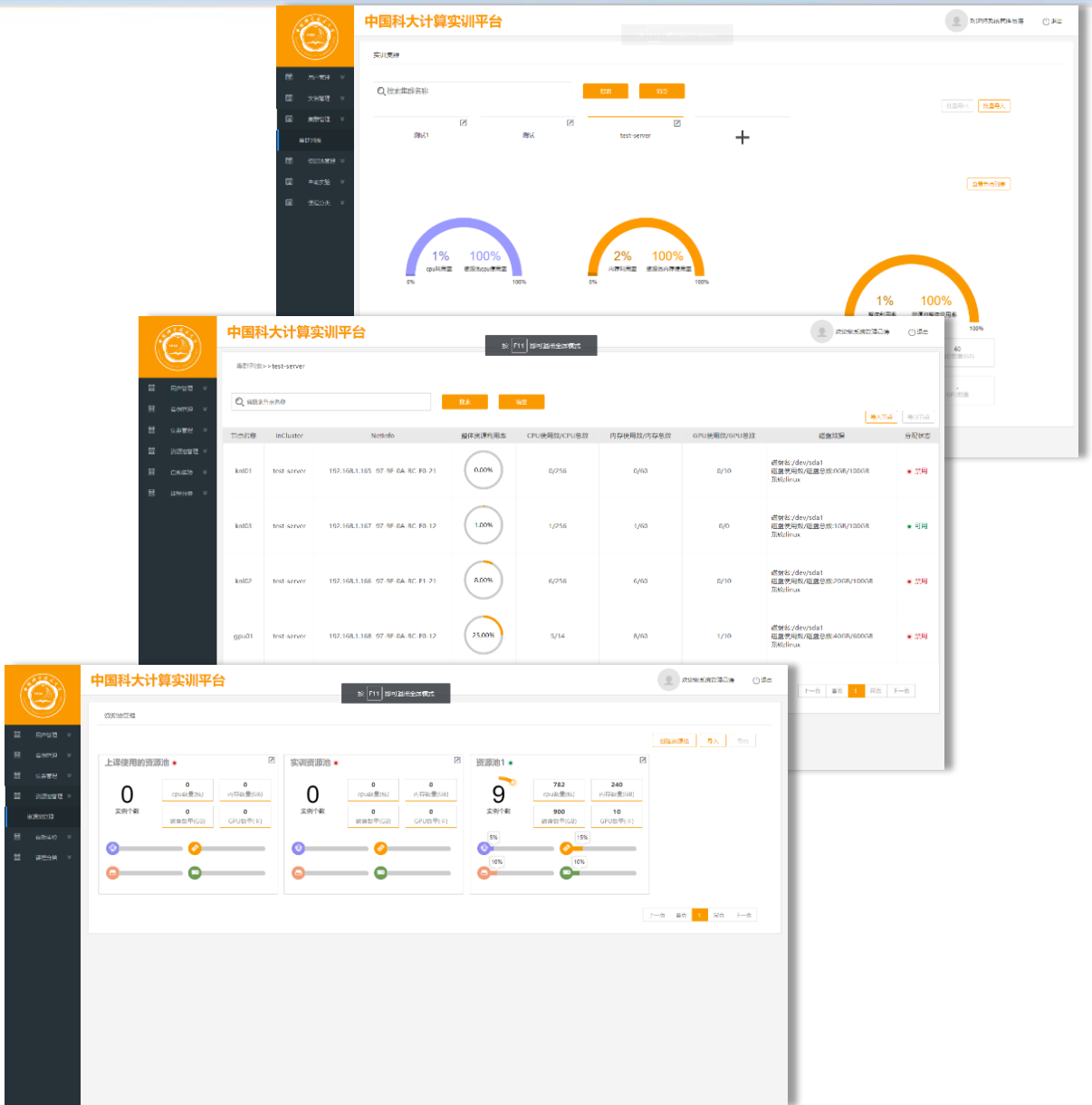
教师视角

中国科学技术大学
University of Science and Technology of China



功能	价值
自定义课程	可以根据需要优化已有课程和自己定义新的课程
自定义环境	可以为自己的课程定义复杂的环境设置，预装软件和上载数据，当学员上课时会使用教师定义的一致环境
导入已有环境	可以根据自己课程的需求导入厂商提供的环境安装包，让学员使用教师指定的一致环境
编写实训教学指南	可以编辑图文教学指南，并利用系统丰富的提示、引导功能构建生动的全新实训课程
预约上课资源	可以提前预约一次上课时要使用的环境资源，避免上课时学员众多而找不到实训环境
教学数据统计	可以查看教学平台中的教学统计数据，针对性的加强课程内容或作为教学汇报数据
创建集群级、GPU等复杂教学环境	可以创建复杂环境完成全新课程的设计

管理员视角



功能	价值
自动化管理学院现有设备	纳管设备，并在系统中设定设备用途，之后系统便会按照规划用途自主管理环境的使用。
管理系统内实例	管理员可以管理和操作系统内所有实例的生命周期，对异常实例进行处理
集中管理审批申请	可以集中管理学员使用环境的申请，提升学院学习和实验环境的利用率
人员与组织结构管理	规划和管理系统内的人员和组织结构
数据统计与分析	管理员可查看系统资源、实训课程的利用情况，识别资源瓶颈，统计系统活跃情况，为扩容提供数据依据

使用案例：深度学习算法学习

中国科学技术大学 University of Science and Technology of China

反向传播算法详解

结束实训 教材列表 延期实训 剩余时长 02 小时 58 分钟 系统管理员涛

按 F11 即可退出全屏模式

实例列表

实训过程

1 步骤一 考察单个神经元的行为

步骤1 理解梯度下降法：用两种方法,优化成本函数$Loss(y)=\frac{1}{2}(y-y_0)^2+y^2$,先在某点向四周尝试性地跨出一小步,计算成本函数的变化,然后从此系列变化中总结出最佳的"跨步"方式,即总结出梯度下降法。

提示：用一个简单的例子理解为什么梯度下降法有效。

假设成本函数为

$$Loss(y)=\frac{1}{2}(y-y_0)^2+y^2$$

其中$y0=[1.0,2.0]$

当$y=[0.5,0.2]$时,我们可以计算成本函数的值。

动作1-1 计算原函数的值 mark

1-1 动作： 计算原函数的值,打开空白python文档task1.py。在task1.py粘贴下面代码。(注意：也可在python命令行直接输入动作1-1到动作2的代码)

```
import numpy as np
y0=np.array([1.0,2.0])
#成本函数定义
def Loss(y):return sum(0.5*(y-y0)**2+y**2)
y=np.array([0.5,0.2]) #初始化y
L0 =Loss(y)
print(L0)
```

反向传播算法详解

task1.py

保存 复制 剪切

```
import numpy as np
y0=np.array([1.0,2.0])
#成本函数定义
def Loss(y):return sum(0.5*(y-y0)**2+y**2)
y=np.array([0.5,0.2]) #初始化y
L0 =Loss(y)
print(L0)

# 尝试四个方向的小挪动
ddy = np.array([[0.001,0],[-0.001,0],[0,0.001],[0,-0.001]])
# 更新y
ynew = [y+ddy[0],y+ddy[1],y+ddy[2],y+ddy[3]]
# 更新 Loss
Lnew = [Loss(ynew[0]),Loss(ynew[1]),Loss(ynew[2]),Loss(ynew[3])]
print('原成本：', L0)
print('更新后的成本：', Lnew)
# 计算成本函数的上升与下降,挪动后的结果减去原结果。
print('成本变化值：', Lnew-L0)
```

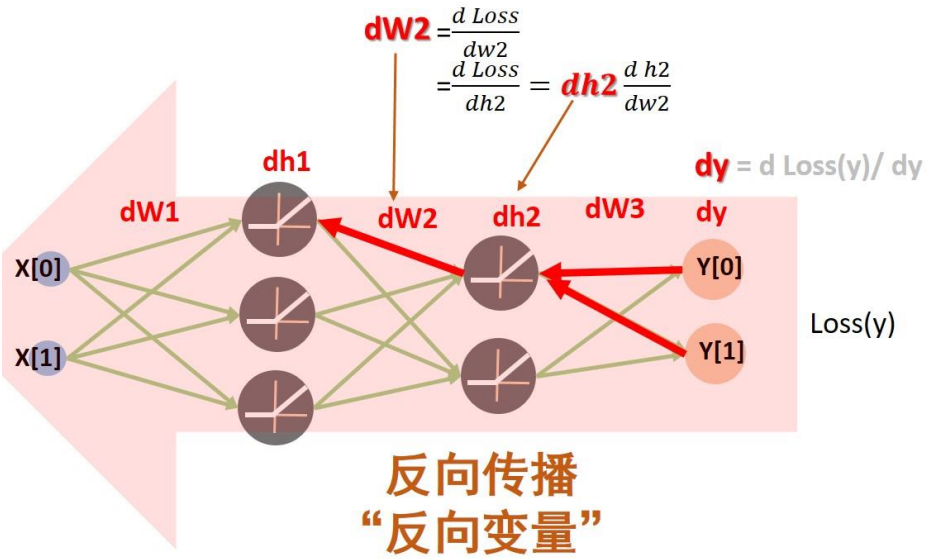
10.244.2.249

命令行1 + 清屏

```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

litao@ba1a3874c514:~$
```

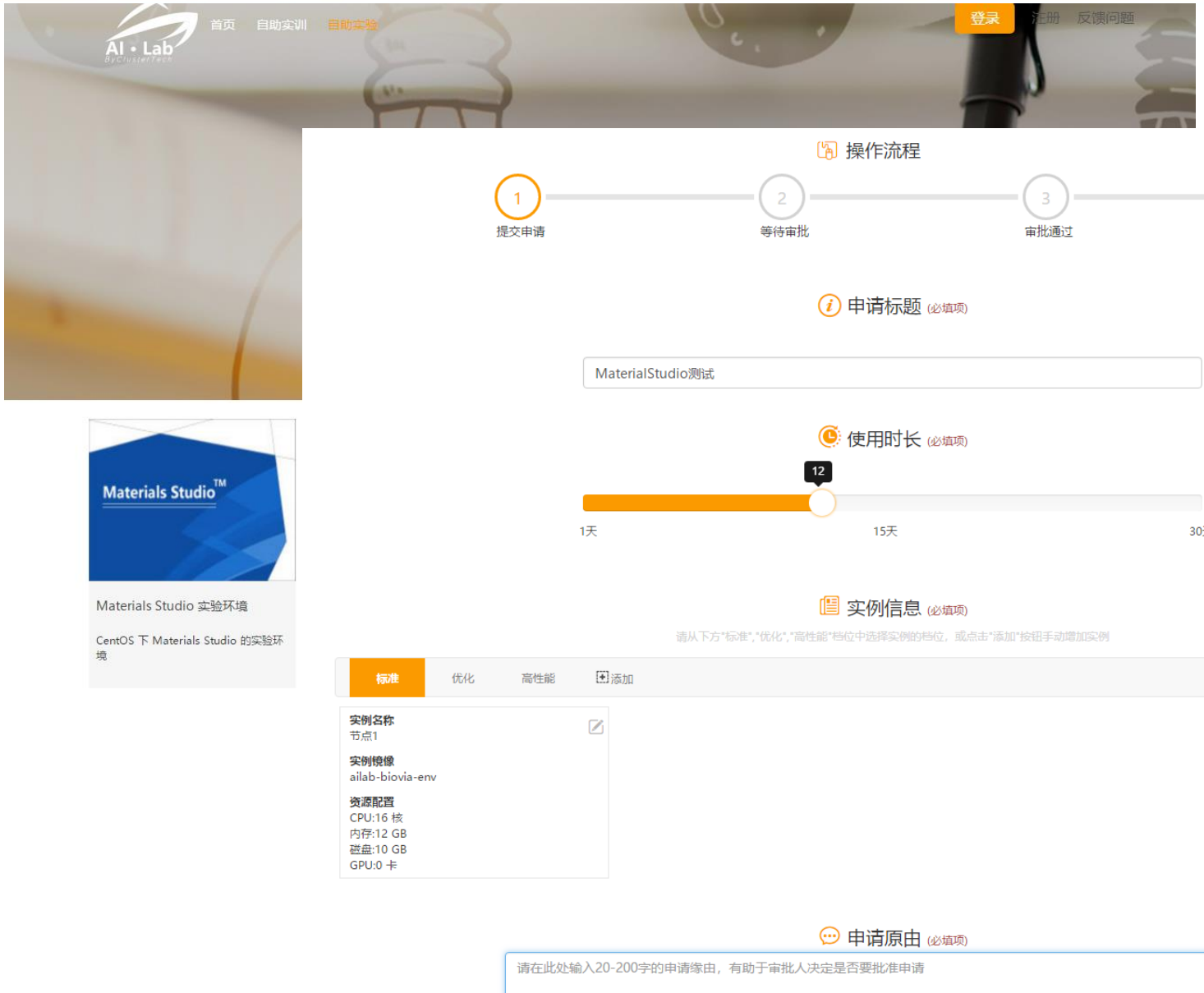
- bin
- boot
- dev
- etc
- home
- lib
- lib64
- media
- mnt
- opt
- proc
- root
- run
- sbin
- srv
- sys
- tmp
- usr
- var
- export
- lesson_2
  - drawNN.py
  - task1.py
  - task2.py
  - task3.py
- run_jupyter.sh
- test
- notebooks
- tensorflow-1.4.1-cp3...

192.168.1.164/admin/.../task1.py','desc':'打开程序文件','instance_name':'反向传播算法详解 %7D



计算顺序：Loss-> dy -> dW3 -> dh2 -> dW2…

使用案例：Materials Studio实验



实训环境

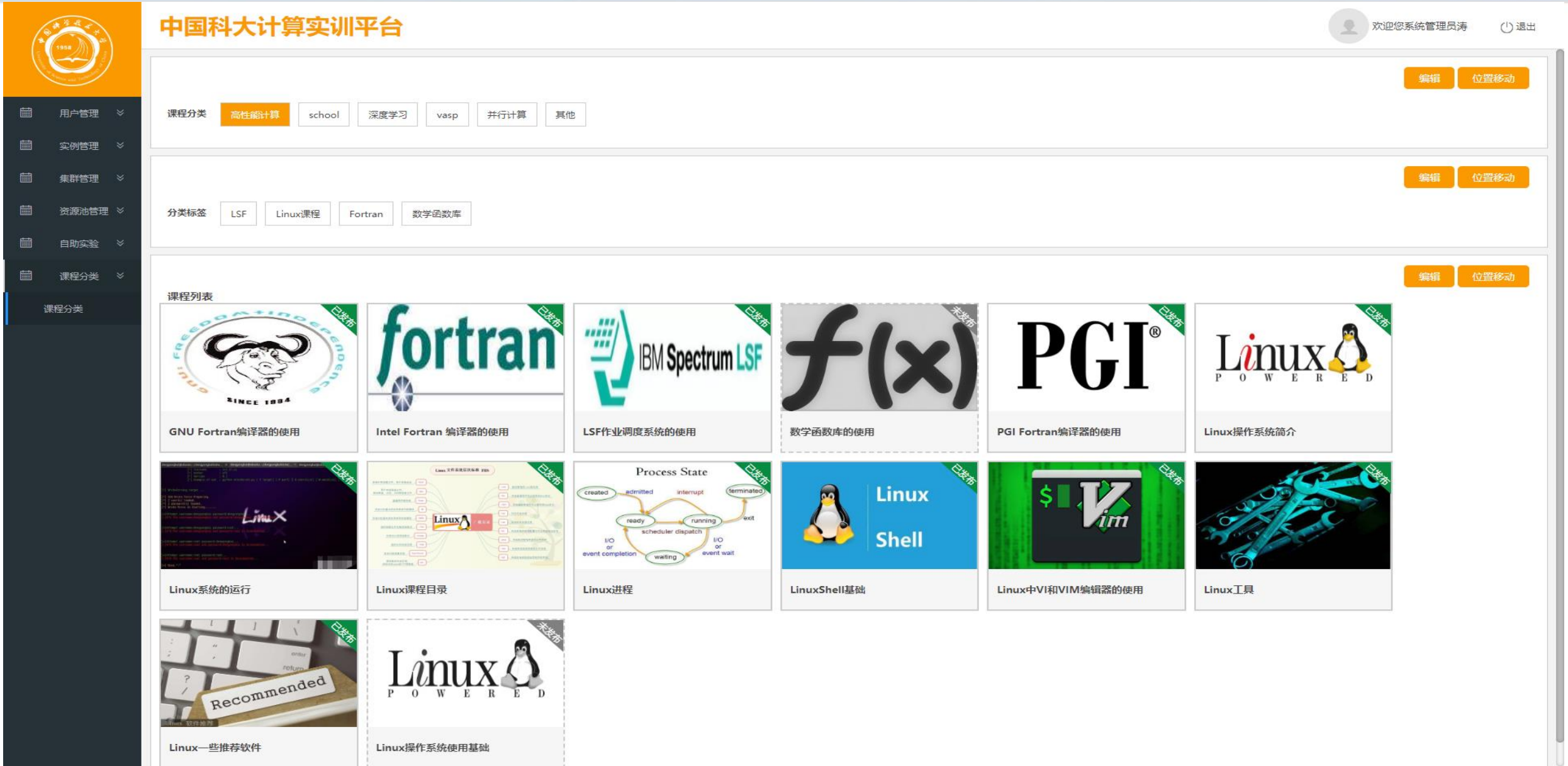
- 配置16核、10G内存
- 所有环境完全相同

实验环境

- 最大64核、64GB内存
- 最大使用30天
- 性能超过Xeon 6142

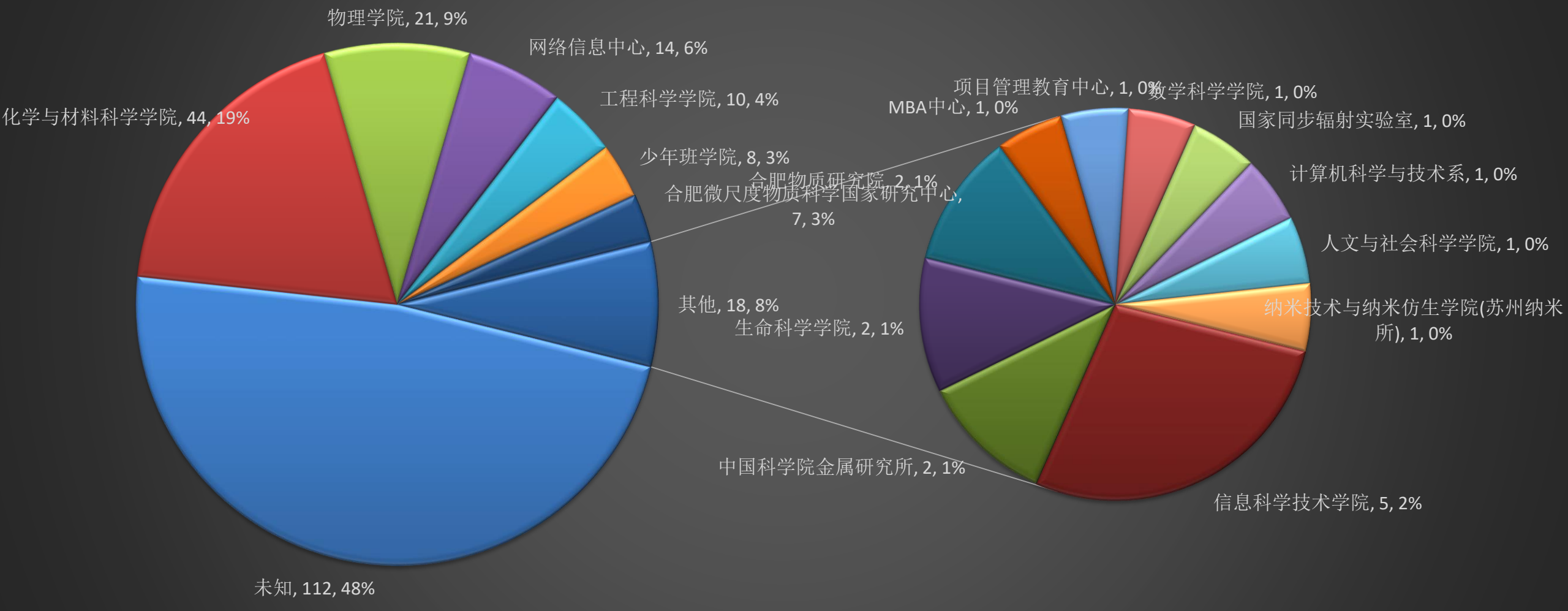
使用案例：HPC使用培训

中国科学技术大学
University of Science and Technology of China

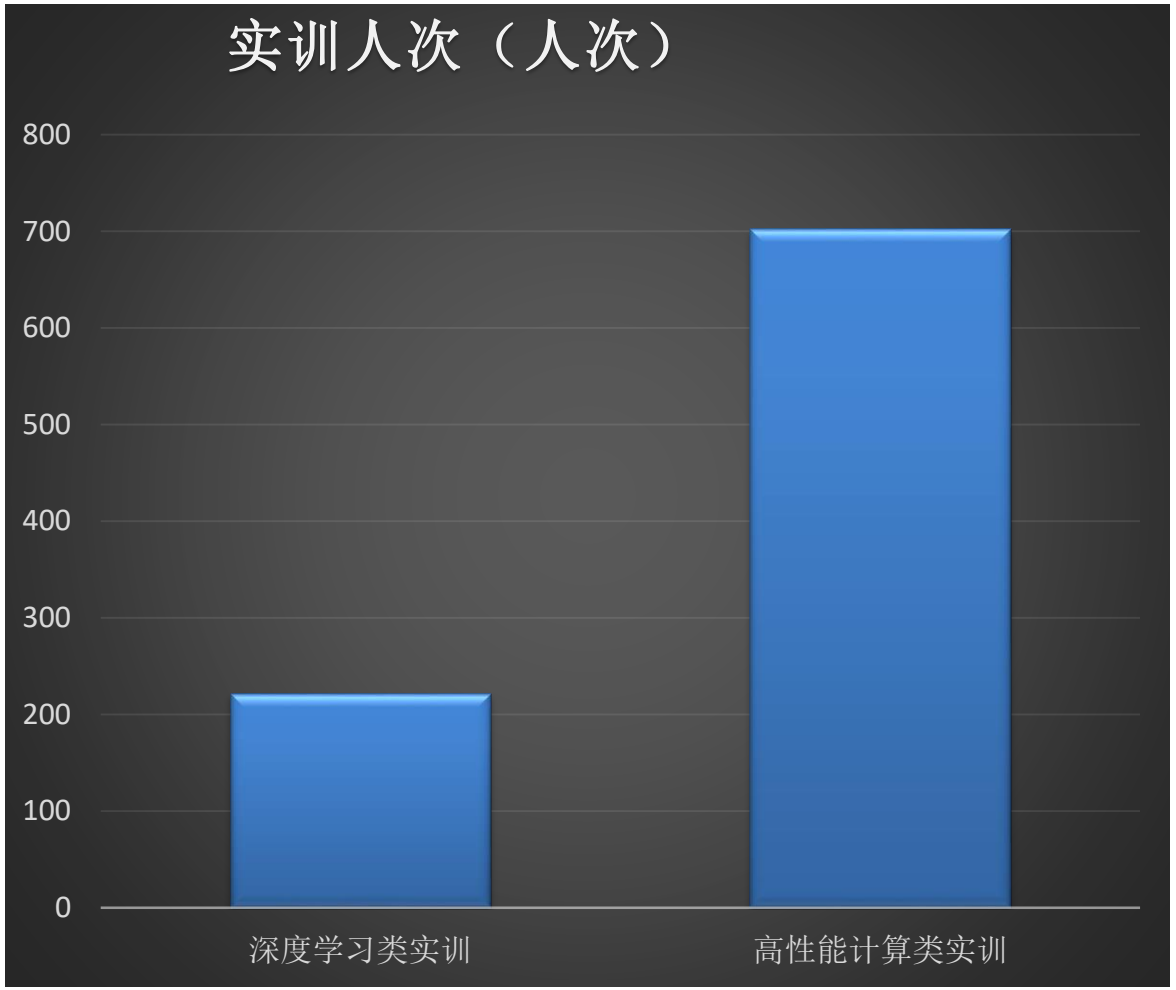


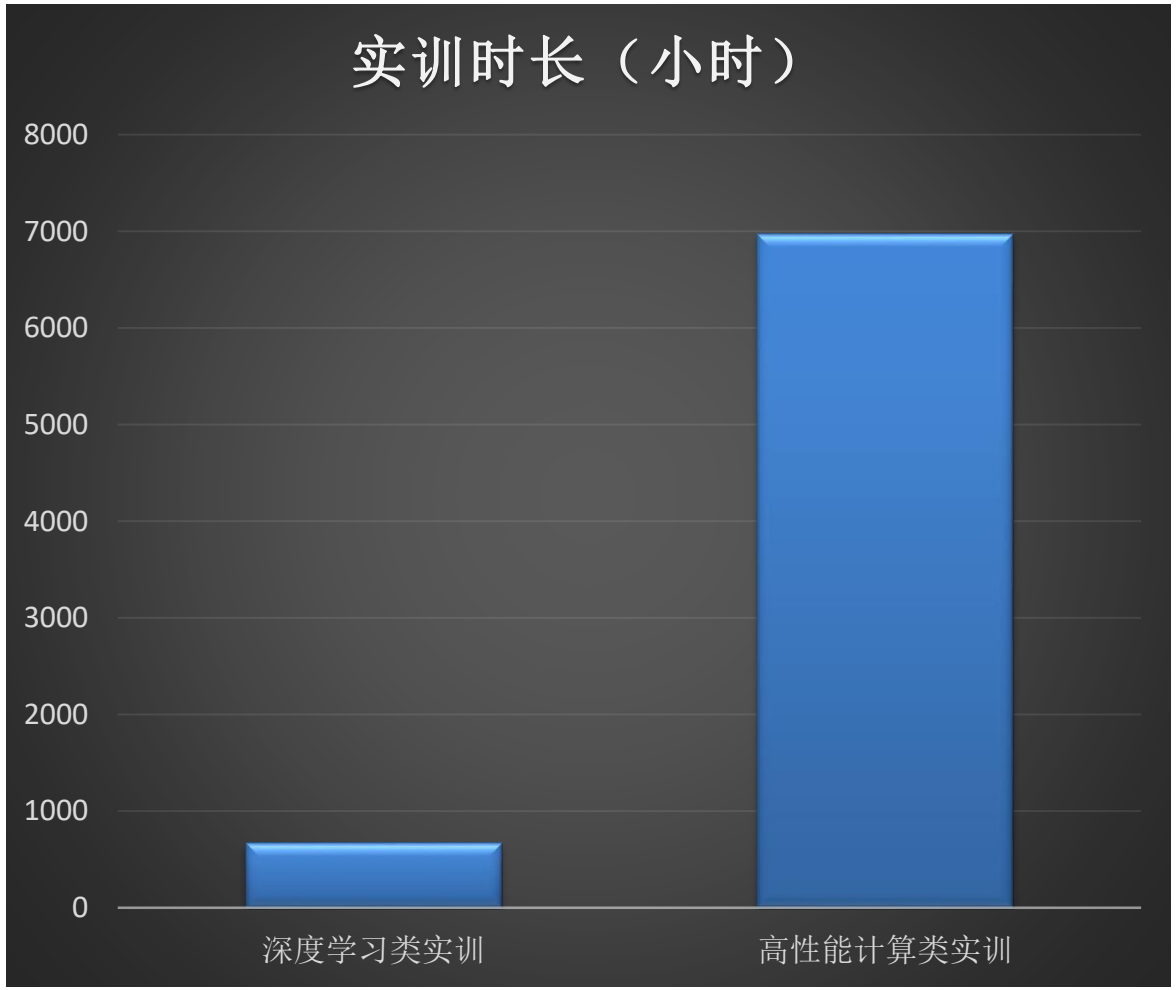
- 为什么要建？
- 系统架构及功能
- 运行数据
- 未来展望

用户单位统计


物理学院, 21, 9%
网络信息中心, 14, 6%
工程科学学院, 10, 4%
少年班学院, 8, 3%
合肥物质研究院, 2, 1%
合肥微尺度物质科学国家研究中心, 7, 3%
化学与材料科学学院, 44, 19%
其他, 18, 8%
生命科学学院, 2, 1%
中国科学院金属研究所, 2, 1%
未知, 112, 48%
项目管理教育中心, 1, 0%
数学科学学院, 1, 0%
MBA中心, 1, 0%
国家同步辐射实验室, 1, 0%
计算机科学与技术系, 1, 0%
人文与社会科学学院, 1, 0%
纳米技术与纳米仿生学院(苏州纳米所), 1, 0%
信息科学技术学院, 5, 2%

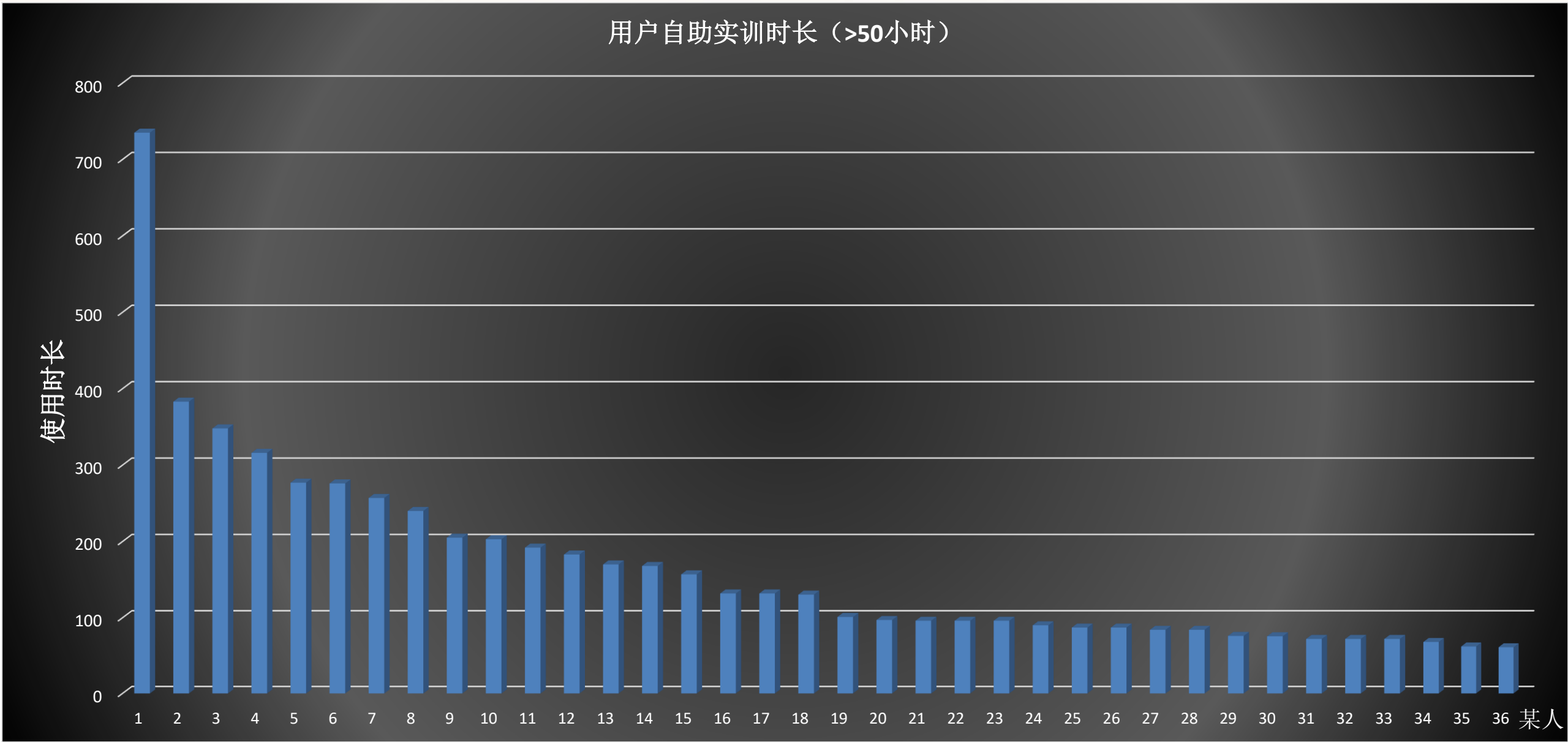
自助实训统计


实训人次（人次）
800
700
600
500
400
300
200
100
0
深度学习类实训
高性能计算类实训

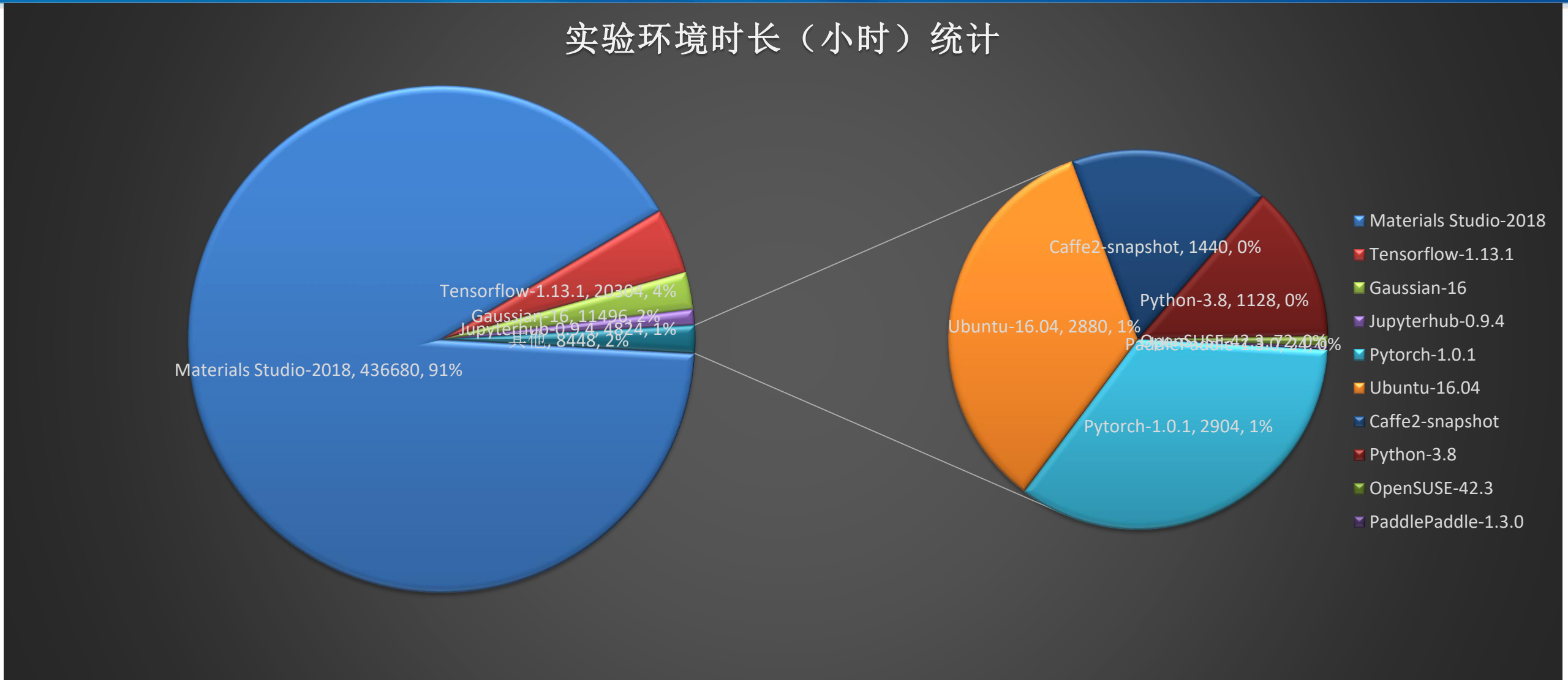

实训时长（小时）
8000
7000
6000
5000
4000
3000
2000
1000
0
深度学习类实训
高性能计算类实训

自助实训用户使用量（>50小时）

中国科学技术大学
University of Science and Technology of China



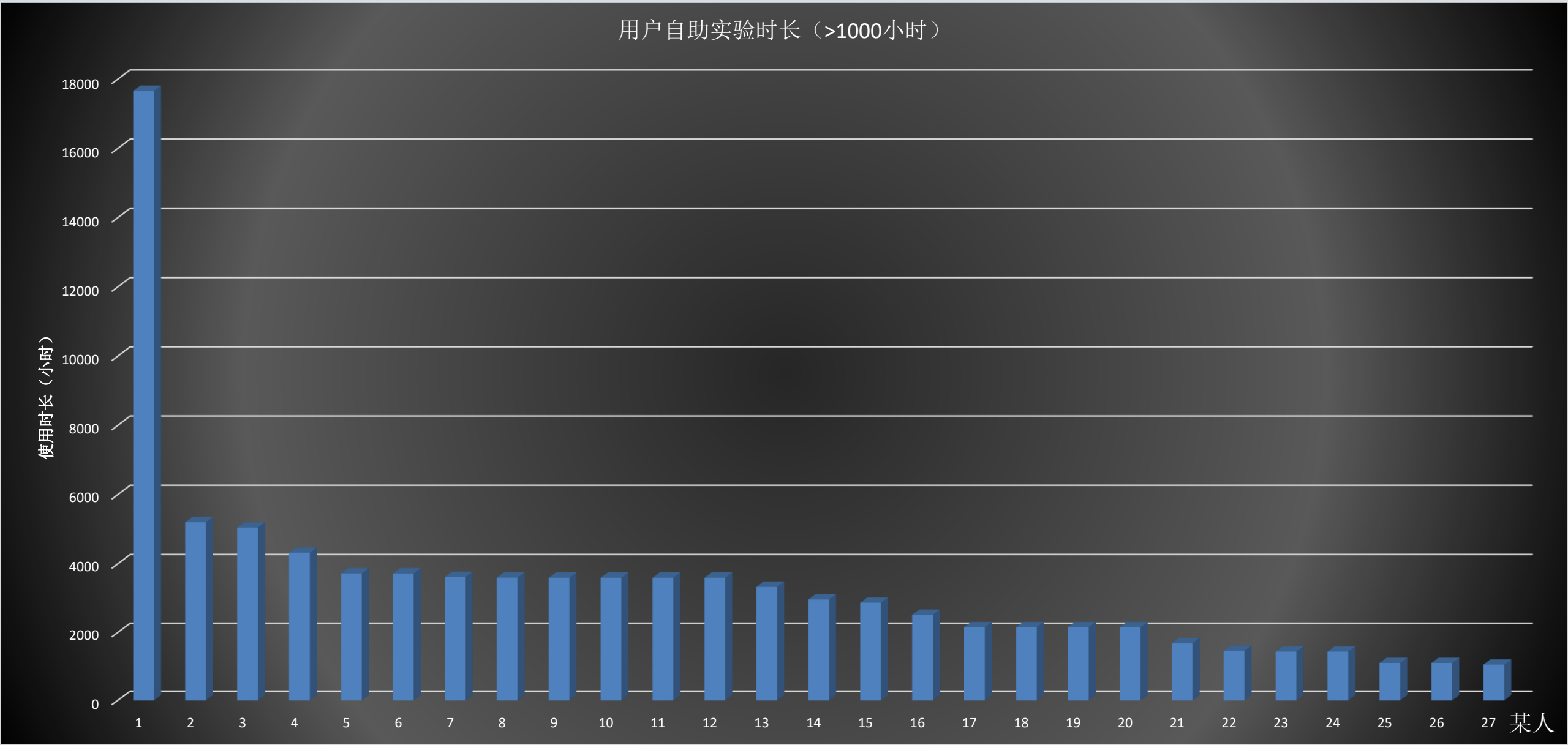
自助实验环境统计



支持了Materials Studio材料科学软件教学活动，其环境占据了主要部分

自助实验用户使用量统计

中国科学技术大学
University of Science and Technology of China



- 为什么要建？
- 系统架构及功能
- 运行数据
- 未来展望

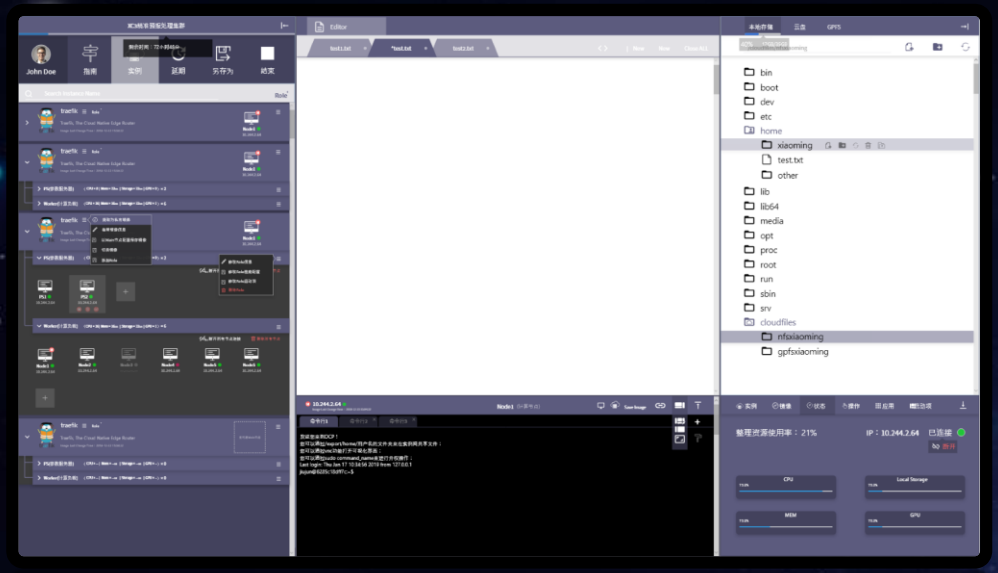
镜像可视化构建

Dockerfile编辑　可见层　只读层　环境作用域

容器内权限　容器安全　访问控制　容器内挂载

网络配置　端口穿透

VS

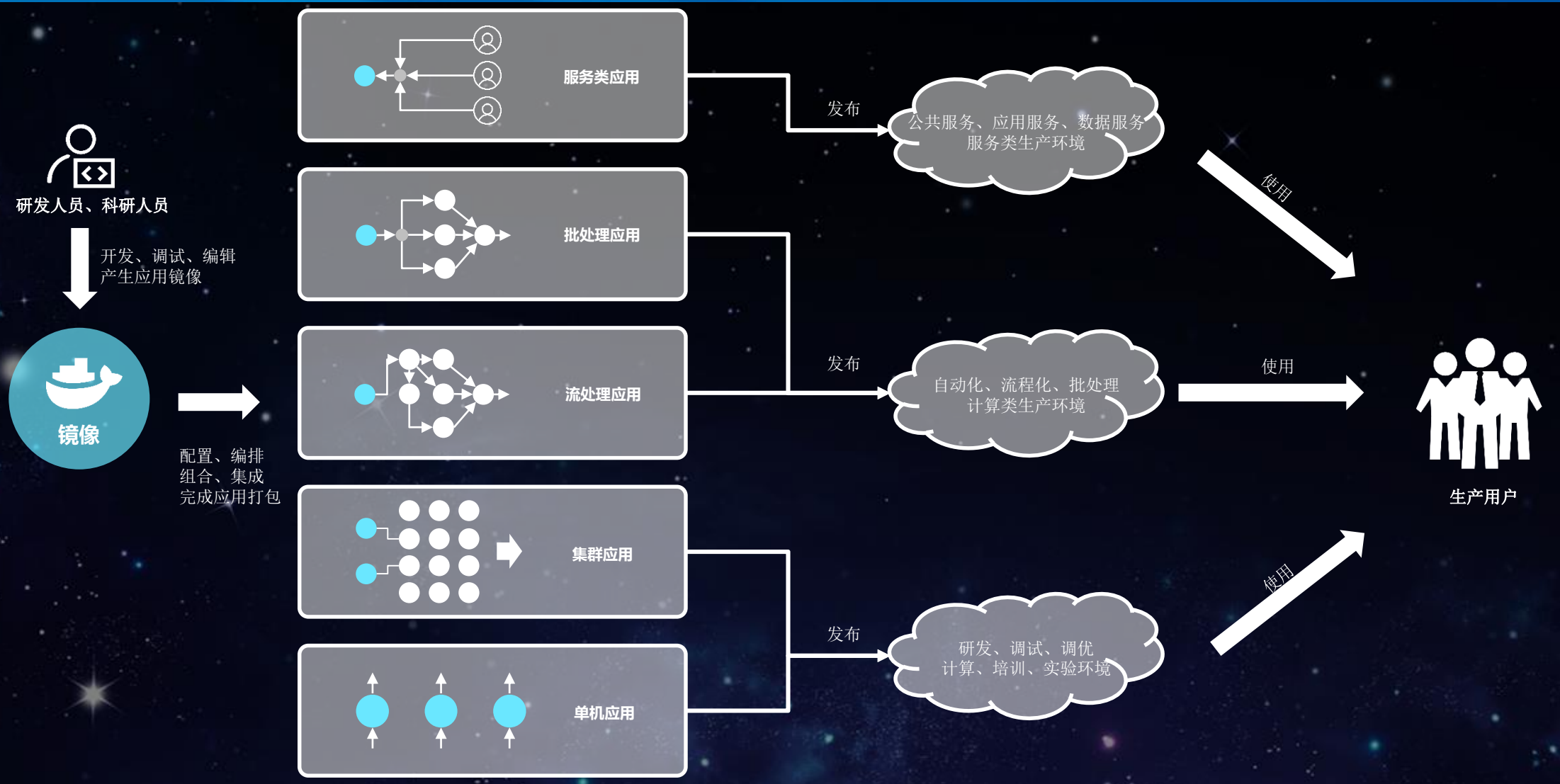


全程可视化，所见即所得构建，无须专业知识，只需一键保存

便捷的所见即所得打包过程

- 保持技术人员原有使用习惯，不需要学习任何容器镜像构建概念，全程可视化，所见即所得构建方式。
- 完整编排能力，可在平台上构建复杂集群级应用系统，兼顾AI应用支持GPU容器构建。
- 构建、发布、使用 全生命周期支持，不改变任何工作习惯。
- 用户就像在传统集群上开发、配置一样，操作后直接形成镜像。

支持更多场景



生产	传播	使用

优化使用体验

中国科学技术大学
University of Science and Technology of China

神经网络入门之全联接网络

指南 | 实例 | 数据集 | 环境变量 | 启动项

基本信息 | 任务 | 教材

任务列表 3 of 25

1. 步骤一 搭建Tensorflow环境
2. 步骤二 构建全联接神经网络模型
3. 动作1 打开文件MNIST_FC.py
4. 动作2 导入所需要的各个库
5. 动作3 定义整个网络的输入
6. 动作4 定义权重与偏置
7. 动作5 计算第一层网络
8. 动作6 步骤检查
9. 步骤三 构建二、三全联接层
10. 动作1替换恰当数字
11. 动作2 替换恰当数字
12. 动作3 输出
13. 动作4 检查
14. 步骤四 定义交叉熵训练器
15. 动作1 定义交叉熵
16. 动作2 定义测试精度
17. 动作3 检查
18. 步骤五 编写训练过程
19. 动作1 定义遥控器sess
20. 动作2 输入训练集
21. 动作3 输出整个测试集的预测精度
22. 动作4 输出模型参数
23. 动作5 检查
24. 步骤六 进行训练，考察最终训练精度
25. 动作1 运行模型

动作1 打开文件MNIST_FC.py

2-1 动作:打开空文件MNIST_FC.py

上一步 下一步

Node_0

MNIST_FC.py

新建文件 保存

```
#end act 2
#start act 3
#start act 3


#代码改写处5  要求第二层的神经元数目为50
W02 = weight_variable([*,*],'W02')      #定义 二层网络 全连接 权重 张量
b02 = bias_variable([*],'b02')           #定义 二层网络 神经元的 偏置 数组
x02 = tf.nn.relu(tf.matmul(x01, W02) + b02)#定义 二层网络 神经元的 激活函数

#代码改写处6  要求输出层的神经元数目为10
W03 = weight_variable([*, *],'W03') #定义 输出层网络 全连接 权重 张量
b03 = bias_variable([*],'b03')       #定义 输出层网络 神经元的 偏置 数组
h03 = tf.matmul(x02, W03) + b03
y =   **** #代码补全处7  提示用tf.nn.softmax函数，h03的维度为[-1,10]

#end act 3
#start act 4

#代码粘贴处8
```

10.244.0.192
镜像最后修改时间：2019-10-15 16:02:41

Node_0 (Node)

命令行1

```
 * Support:        https://ubuntu.com/advantage

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

xinwei@aa8cccfab09f:~$
```

本地储存

路径:/ 显示隐藏文件 root权限

- bin
- boot
- dev
- etc
- export
- home
- lesson_4
  - MNIST_FC.py
  - MNIST_data
  - chkcode
  - data
- lib
- lib64
- media
- mnt
- notebooks
- opt
- proc
- root
- run
- run_jupyter.sh

角色: Node　索引: 0

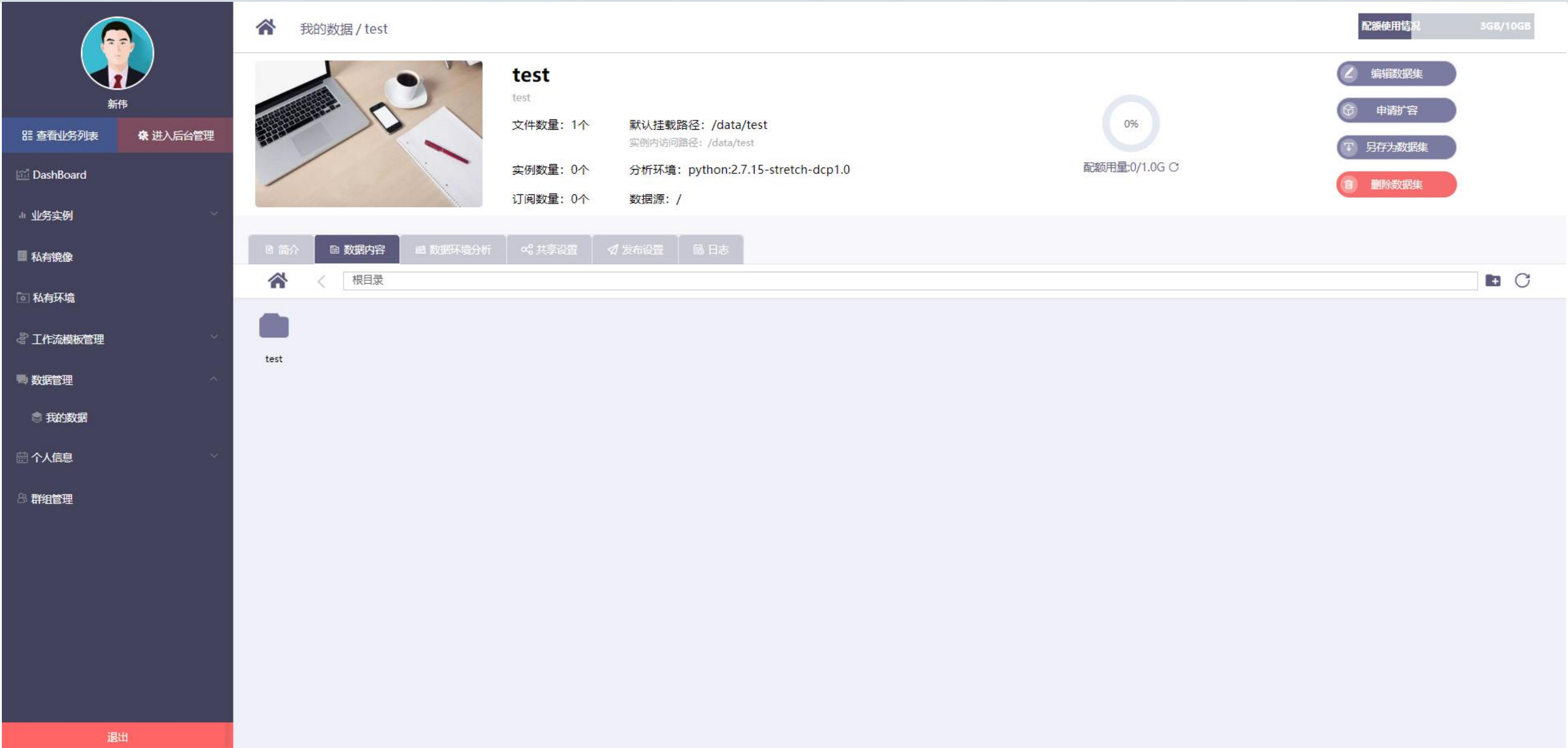
实例名称: Node_0 (Main)　别名: 神经网络入门之全联接...

CPU核心数: 4 Core　内存容量: 5 GB

GPU数量: 0 Card　本地存储数量: 5 GB

数据共享机制

中国科学技术大学
University of Science and Technology of China



应用流程化支持

中国科学技术大学
University of Science and Technology of China

范瑾

查看业务列表 | 进入后台管理

- DashBoard
- 业务实例
- 私有镜像
- 私有环境
- 工作流模板管理
- 数据管理
- 个人信息
- 群组管理

退出

工作流模板

test

工作流的真实名称：mgt:5000/dcp/fanjin/workflow/9d52f5dc-a894-43e1-b241-67f0a3a5c3e3:v1.0.0

创建模板副本 | 删除此模板 | 发布此模板 | 创建运行实例

基本信息 | 简介 | 数据集设置 | 共享设置 | 工作流配置 | 运行控制设置 | 参数控制 | 日志

```yaml
# API的声明，使用的是 argoproj.io的自定义组的API
apiVersion: argoproj.io/v1alpha1
# 分类是WF
kind: Workflow
# 元数据描述
metadata:
# 生成名称
  generateName: hello-world-
# 规格说明
spec:
  # 入口
  entrypoint: whalesay
  # 模板说明
  templates:
    # 名称
  - name: whalesay
    # 实例说明
    container:
      # 所用镜像
      image: docker/whalesay:latest
      # 执行的命令，关于这个cowsay是怎么回事可以参考
      # https://blog.csdn.net/chengqiuming/article/details/79211395
      command: [cowsay]
      # 传递的参数
      args: ["hello world"]
```

查找参数 | 查找镜像

查找镜像

查找历史

取消 | 保存

中国科学技术大学
University of Science and Technology of China

谢谢！

中国科大超级计算中心

http://scc.ustc.edu.cn