

# 网络与服务器使用规范（2020-09-17修订版）

## Part 1 实验室高性能计算集群（HPC）简介

集群拓扑结构：

用途	节点名称	配置信息
登陆节点	master	不作计算用途
CPU计算节点	amd01	24核+32G
同上	pan01	64核+1536G
同上	node02	44核+384G
同上	pan02	80核+1024G
GPU计算节点	gpu01	20核+128G+3*Nvidia显卡

访问方式：ssh+节点名称，如 ssh pan01

## Part 2 PBS简单教程

### 2.1 PBS简介

作为集群系统软件的重要组成部分，集群作业管理系统可以统一管理和调度集群的软硬件资源，保证用户作业公平合理地共享集群资源，提高系统利用率和吞吐率。常用的集群作业管理系统：PBS（Portable Batch System），LSF 和 SLURM，其中PBS主要包括openPBS, PBS Pro和Torque三种分支。本机采用了PBS系统Torque版本。[PBS使用指南](#)

### 2.2 PBS简单使用示例

- 1. 查看资源占用情况  
命令： pestat  
节点状态
  - excl：所有CPU资源已被占用；

- busy : CPU已接近满负荷运行;
- free : 全部或部分CPU空闲;
- offl : 管理员手动指定离线状态。

如需要获取精确的节点核心占用情况, 需使用 `qstat -f` 或 `pbsnodes` 命令获取具体节点具体核心的占用。

## 2. 准备任务shell脚本

脚本开头添加PBS相关参数, 指定完PBS参数后再添加相关shell脚本代码。

```
#PBS -N kggseq@pan02
#PBS -l nodes=pan02:ppn=60
#PBS -l mem=200G
#PBS -o output.log
#PBS -e error.log
#PBS -q default

# 要执行的shell脚本代码
java -Xmx10g -jar ./kggseq.jar
```

### 参数解读

- jobname: 指定提交的作业名 (如kggseq@pan02)
- nodes: 指定任务运行节点 (如pan01、pan02等)
- ppn: 申请的CPU核数
- mem: 申请的内存大小 (暂统一以G为单位)
- output.log: 存放输出日志文件路径 (自行指定)
- error.log: 存放错误日志文件路径 (自行指定)
- default: 提交至默认目标队列 (无需修改)

## 3. 提交任务

建议在master节点上, 执行qsub提交上述准备好的脚本, 如 `qsub qsub_kggseq.sh`

## 4. 任务状态查询

`qstat` 查询任务状态, 任务状态为“R”代表正在运行, “H”代表等待可用资源, “C”代表异常终止

`qstat -f 任务号` 查询任务详情

# Part 3 软件与环境

## 1. 背景

HPC软件和环境管理主要基于conda和environment modules进行。conda是开源包 (packages) 和虚拟环境 (environment) 的管理系统, [介绍链接](#)。

Environment Modules 是一个简化 shell 初始化的工具, 它允许用户在使用 modulefiles 进行会话期间轻松修改其环境, [介绍链接](#)。

基于conda优越的包管理能力, 我们推荐采用conda进行R/Python版本和包管理, [conda使用教程](#)。

## 2. 概况

- conda背景：对于R/Python而言，已经安装R3.3/3.4/3.5/3.6四个版本，Python2.7/3.7两个版本供大家使用（环境路径位于/app/sys/miniconda3/envs），可通过 `conda info --envs` 查看，通过 `conda activate <环境名>` 激活相应环境。在上述环境下装包方式与之前相同，**暂不支持conda方式装包**（方便统一管理）。若如有特殊需求，请**自行构建环境**。
- module背景：为方便一些其他常用软件版本和环境的管理问题，同时采用Environment Modules进行管理。例如kggsum，已将其依赖环境写入modules管理，在使用前执行 `module load kggsum` 即可导入依赖环境。用户可使用 `module avail` 查看可使用的模块。如果有环境和版本管理需求，**请与管理员龙奇涵联系**。[module使用教程](#)。

**以上所有软件，包和环境设置均在HPC所有节点共享。**

### 3. 使用指南

- 如果使用新的R/Python环境，需要**重新装包**，如果版本相同，亦可直接引用或移植。过渡期间，之前的R/Python环境仍然保留。
- 使用上述软件前请先执行`source /etc/profile`（仅需执行一次即可）。**首次**使用conda时，须先执行一次 `source activate`，否则可能报错。
- 如果 `conda info --envs` 看不到上述R/Python环境，请在 `$HOME/.condarc` 中加入：

```
envs_dirs:
- /app/sys/miniconda3/envs
```

也可以自行添加相关环境配置，详见[.condarc配置文件格式](#)。

### 4. 用户软件安装示例

- R: `install.packages("package_name", lib="用户指定的库存放文件夹")`
- Python: `pip install --user package_name`

### 5. conda笔记

- [正确解决](#) `CommandNotFoundError: Your shell has not been properly configured to use 'conda activate'`.
- [安装指定版本/build](#)
- [环境加载路径，镜像设置等](#)

## Part 4 普通用户规范

### 1. 软件安装：

- 可执行程序、python/R包安装，存放到自己目录，如需共享，可存放至/sdb1/tools;
- 编译安装软件时需要相关系统依赖库时请联系**朱正、龙奇涵**安装。

### 2. 任务提交：PBS提交系统，合理使用资源。PBS问题联系**朱正、龙奇涵**。

### 3. 系统故障，联系**朱正**。

### 4. 项目数据存放，文件共享：联系**彭文杰（薛超）**。

### 5. 磁盘空间限制8TB，超过后将被限制登陆，联系**彭文杰（薛超）**。

### 6. 临时目录路径：/public1/tmp，30天后将被清除。

7. 网络使用问题，联系**彭文杰**、**薛超**解决。