

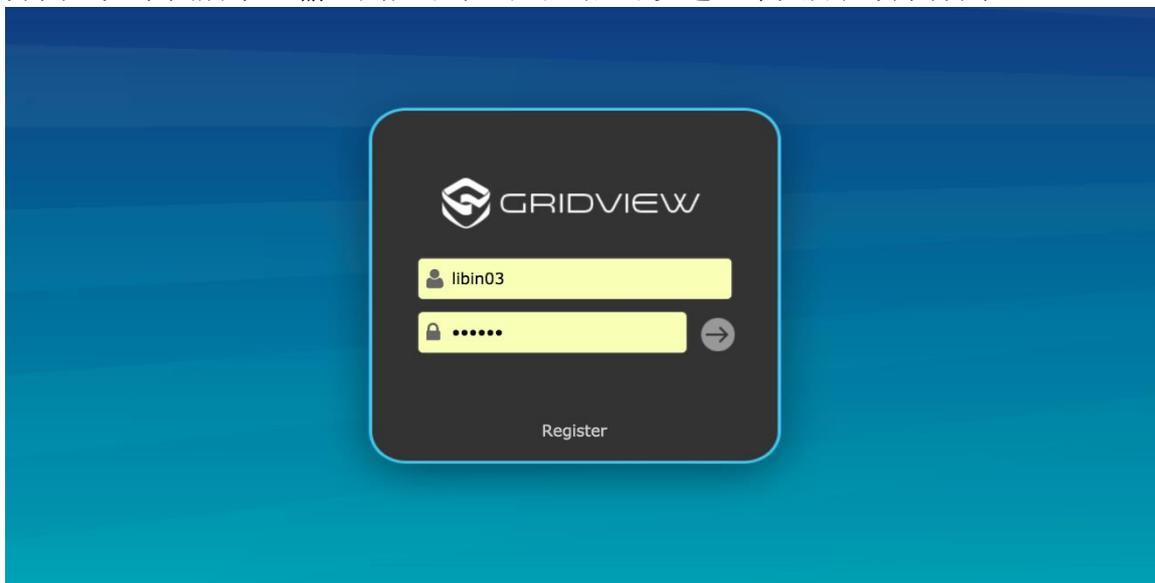
Nebula 生信超算平台使用指南

1. 服务器登录和常见用法

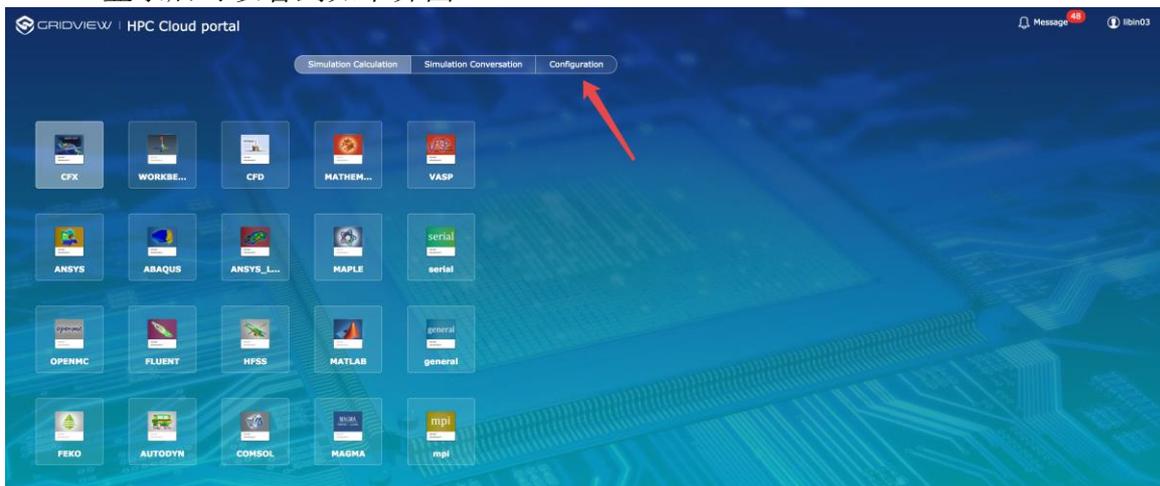
登录方式一般分为两种，一种是使用 Web 浏览器登陆 Gridview，另外一种方法是使用 SSH 远程登陆服务器。Web 浏览器登录 Gridview 后可以查看用户个人在上个时间段（日、周、月）的使用计算机时的情况；而 SSH 远程登陆服务器可以操作个人文件目录，并提交作业进行计算。此外可以选择支持 SFTP 的软件（如 FileZilla）登陆服务器进行文件传输。

1.1 Gridview

在 Web 浏览器网址栏中输入 <http://nebula.ustc.edu.cn:6080> 进入 Gridview 界面，如下图所示，输入用户名和密码之后可以进入个人账户界面目录。



登录后可以看到如下界面：

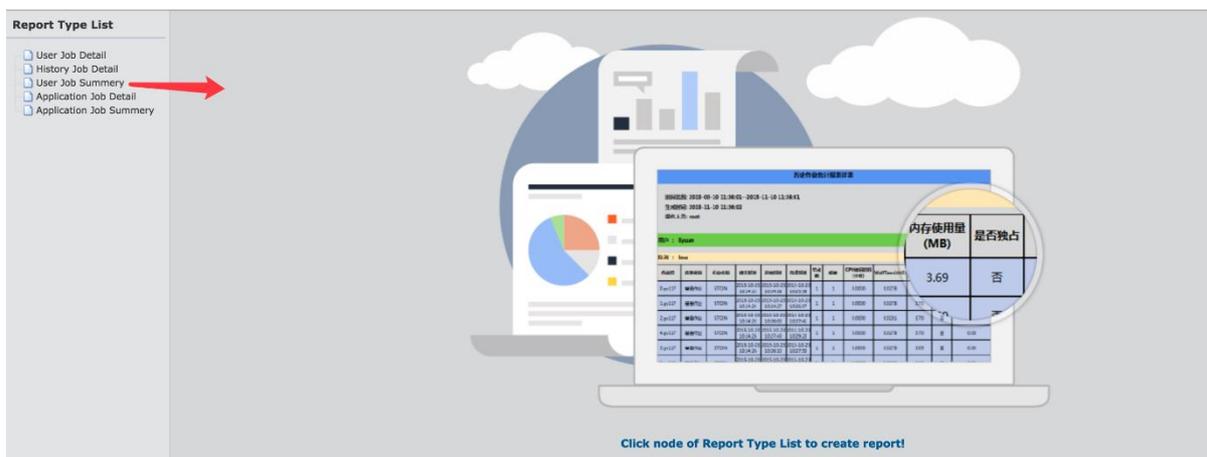


如上图箭头所示，要查看历史完成作业情况，请点击主界面中的 Configuration 选项。

进入 Configuration 界面后如下图所示：



在该界面下进入 Job Report 版块，可以看到如下图所示选项：



点击进入 User Job Summary 后看到如下图界面选项。

Report Type	User Job Summary
Report time config	
Time type:	<input checked="" type="radio"/> Predefine <input type="radio"/> Custom define
*Time span	last hour
Basic info filter	
User name	libin03
	Precise
Create report	

在 Time span 选项中可以选上一日、上周、上个月、上个季度或者上一年，作为查询用户所用机时的时间范围，也可以在 Custom define 选项下自定义时间范围，然后点击 Create report 则会生成工作汇总表，如下图所示：

用户作业统计报表总表

时间范围: 2017-08-02 00:00:00--2017-08-03 00:00:00

生成时间: 2017-08-03 11:13:31

操作人员: libin03

用户名	队列名	CPU使用时间		WallTime		内存使用量		Efficiency
		数值(小时)	百分比(%)	数值(小时)	百分比(%)	数值(MB)	百分比(%)	百分比(%)
libin03	low	0.0000	0.00	0.0003	0.23	0.00	0.00	0.00
	middle	0.0978	100.00	0.1326	99.77	350.75	100.00	73.76
	总计(按队列)	0.0978		0.1329		350.75		73.59
总计(按用户)		0.0978		0.1329		350.75		73.59

在上图生成的表中，我们可以查看在用户在特定时间段中 CPU 使用时间，WallTime 时间（表示实际使用机群的时间），内存使用量等等。其中 WallTime 时间是用户占用机群资源进行计算的**实际时间，因此将按照这一时间作为用户机时使用的收费依据**。

更多的细节，比如查看单个任务使用的机时，可以在 Job Report 版块下的 History Job Detail 选项查看，查询方法与上面类似。

1.2 SSH 服务器

在 Mac OS X 系统或者 Linux 系统下打开终端(Terminal)，如果是 windows 系统为方便使用可以下载相关软件（如 putty）。然后键入如下命令：

```
ssh <username>@nebula.ustc.edu.cn
```

如：ssh libin03@nebula.ustc.edu.cn

然后提示输入密码（注意：密码内容不会在屏幕上显示）。登录成功后默认位于个人目录下，并可以输入常见 Linux 命令运行(ls、mkdir、cd、vi 等等)，ssh 会将输入的指令传给服务器继而执行，然后在屏幕上显示相应结果，和直接使用服务器 Linux 系统无本质区别。

可以输入 `module avail` 命令查看当下有那些可以使用的软件，如下图所示。

```
[libin03@mgt ~]$ module avail
----- /public/MODULES/COMPILER -----
CUDA8.0 INTEL/parallel_studio_xe_2016.2.181 INTEL/parallel_studio_xe_2017_update4 oracle-jdk
----- /public/MODULES/APPS -----
MATLAB/R2017a
----- /public/MODULES/BIO -----
bedops bowtie2 cufflinks GATK hisat2 hotspot2 IGVTools kentUtils meme picard Relion STAR tophat-2.1.1
bowtie chilin fastqc HiC-Pro HOMER IGV juicer lammps modwt preseq sratoolkit tophat-1.4.1 Trinity
```

在服务器下可以执行 Python、bedtools、samtools 等已经内置好的软件，但是一些软件无法直接在服务器下直接执行，比如 bowtie、bowtie2、picard、STAR

等。在执行这些软件之前首先要输入 `module load <software>` (如 `module load bowtie2`)。如下图所示:

```
[libin03@mgt test]$ bowtie
bash: bowtie: command not found...
[libin03@mgt test]$ bowtie2
bash: bowtie2: command not found...
[libin03@mgt test]$ module load bowtie2
[libin03@mgt test]$ bowtie2
No index, query, or output file specified!
Bowtie 2 version 2.3.2 by Ben Langmead (langmea@cs.jhu.edu, www.cs.jhu.edu/~langmea)
Usage:
  bowtie2 [options]* -x <bt2-idx> {-1 <m1> -2 <m2> | -U <r> | --interleaved <i>} [-S <sam>]

<bt2-idx>  Index filename prefix (minus trailing .X.bt2).
           NOTE: Bowtie 1 and Bowtie 2 indexes are not compatible.
<m1>      Files with #1 mates, paired with files in <m2>.
           Could be gzip'ed (extension: .gz) or bzip2'ed (extension: .bz2).
<m2>      Files with #2 mates, paired with files in <m1>.
```

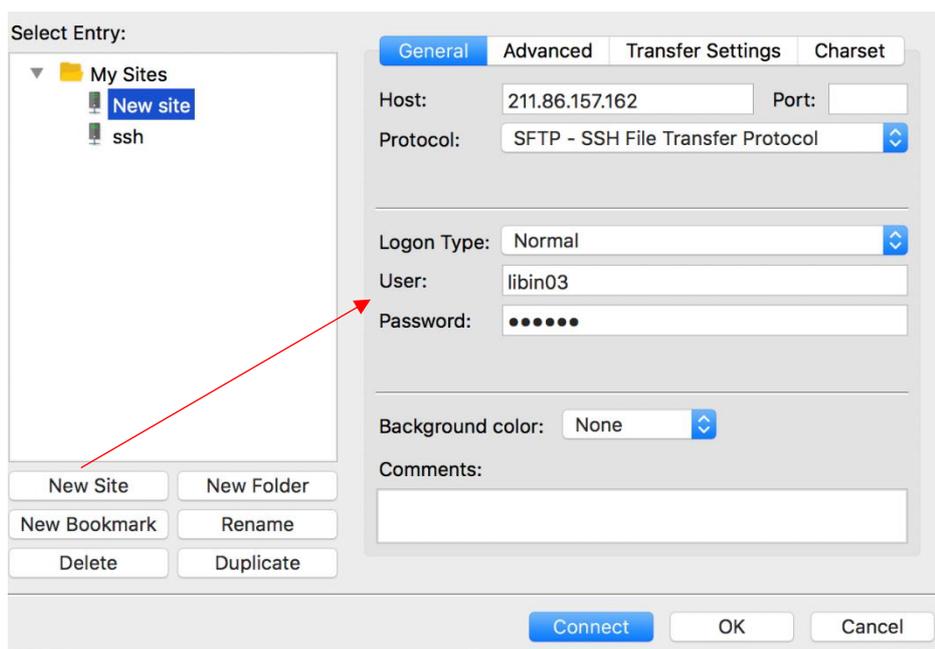
注意: 用户登陆机群之后, 实际上位于**登陆节点** (资源很少), 不能进行大量计算, 不可以直接运行计算的软件和程序, **所有计算任务和作业必须通过 PBS 脚本提交到计算节点** (大量资源) 运行。PBS 脚本编写和提交的方法将在第 2 章节详细描述。

1.3 文件上传和下载

在服务器使用过程中有时候需要上传文件和下载文件, 可以通过 FileZilla 软件快捷解决这一问题。本地打开该软件后可以看到左上角的服务器图标,



点开 after 点击 **New Site** 然后如下图所示填写即可登录个人服务器目录进行上传和下载文件操作。



除此之外还可以通过在终端 (Terminal) 键入 `rsync` 命令上传和下载文件以及软件程序。

2. PBS 脚本

通过 SSH 登录服务器后，**必须通过提交 PBS 脚本的方式提交作业任务或软件到计算节点上运行，不可以在登陆节点上直接计算或运行软件**（大量作业涌入登陆节点会导致系统死机）。下面通过例子来说明相关事项。

2.1 pbs 脚本使用

当一个作业任务需要进行计算，比如编写好的 Python 程序指令或者使用 Bowtie2 软件，必须首先编写 PBS 脚本文件，如下图的 `serial_job.pbs` 脚本文件，（注意：在 PBS 中提交作业时，如果使用某些需要加载的软件，也必须在使用的命令前加入 `module load` 语句预先载入，如下图 PBS 脚本中的倒数第 3 行）。

```
#!/bin/sh
#An example for serial job.
#DO NOT RUN THIS SCRIPT DIRECTLY,
#PLEASE RUN THIS SCRIPT WITH qsub: qsub serial_job.pbs
#
#PBS -N job_name
#PBS -o job.log
#PBS -e job.err
#PBS -q middle
#PBS -l walltime=02:00:00
cd /home/qukun/libin03/test
module load bowtie2
bowtie2 -p 10 -x ./indexes/hg19 -S ./Adim.sam -c ATGCATCATGCGCCAT
python hello.py
```

提交 PBS 脚本时我们一般通过 `qsub <name.pbs>` 命令来完成，如

```
qsub serial_job.pbs
```

当成功提交任务后会有一个 `job.id` 出现在屏幕上。可以通过 `qstat` 命令查看用户个人已经提交并正在运行或者排队的作业信息，相关命令为：

```
qstat -u <username> 如 qstat -u libin03
```

如果在执行任务期间发现错误或者其他原因想要取消任务可以通过执行 `qdel` 命令，形式为：

```
qdel job.id 如 qdel 44.mgt
```

示例如下图所示：

```
[libin03@mgt test]$ qsub serial_job.pbs
44.mgt
[libin03@mgt test]$ qstat -u libin03
mgt:
Job ID          Username      Queue         Jobname        SessID  NDS  TSK    Req'd  Req'd   Elap
-----
44.mgt          libin03      middle        job_name       21465   --   --     --    02:00:00 R  00:00:08
[libin03@mgt test]$ qdel 44.mgt
[libin03@mgt test]$
```

对于使用一个 CPU 的串程序，节点数和线程数默认为 1。而当需要多节点并行程序时就需要用到相关 `mpirun` 命令，一个并行计算的 PBS 脚本文件如下图所示，可以看到在红色细线箭头的指定部分使用了 5 个节点（nodes），每个节点最大使用的核数为 24（ppn，刀片节点不能超过 24，胖节点不能超过 144，GPU 节点不能超过 28），在本次任务中使用了 5*24 共 120 个核进行并行计算。在某些软件使用过程中，比如 `bowtie2`，设置好参数后会自动采用对应的并行参数运行（如并行的 CPU 数），无需在 `bowtie2` 命令下再加入参数。

```
#!/bin/sh
#an example for Intel MPI job.
#DO NOT RUN THIS SCRIPT DIRECTLY,
#PLEASE RUN THIS SCRIPT WITH qsub: qsub intelmpi_job.pbs
#
#PBS -N mpi4py_manybody
#PBS -o job.log
#PBS -e job.err
#PBS -q high
#PBS -l nodes=5:ppn=24
#PBS -l walltime=1:00:00
module load INTEL/parallel_studio_xe_2017_update4
cd /home/qukun/libin03/test_mpi/mpi4py
mpirun python main.py
```

2.2 PBS 脚本参数说明

PBS 脚本可以自己撰写，在服务器上可以直接通过 `vi` 或 `vim` 编辑器打开一个新文件进行编辑，或者 `copy` 别人已经写好的 PBS 脚本然后按照自己的需求来进行修改。

首先介绍 PBS 脚本所定义的运行任务的队列种类，如下表所示：

队列种类	优先级	描述	最大 nodes 数	最大 ppn 数	最大 Walltime
high	高	提交到这个队列的作业高优先级运行，VIP 用户专用	26	28	48:00:00
middle	中	提交到这个队列的作业中优先级运行	26	28	120:00:00
batch	低	用于提交批处理作业，低优先级	26	28	240:00:00
low	低	提交到这个队列的作业低优先级运行	26	28	240:00:00
fat	普通	提交需要一次性占用大量资源（内存和 CPU）的作业	2	144	120:00:00
gpu	专用	提交使用 CUDA 优化和 GPU 资源的作业，GPU 用户专用	1	28	120:00:00

注意！！请勿超过最大 nodes、ppn 和 walltime 的限制！

下面介绍 PBS 脚本中各项参数的定义，一般的 PBS 脚本中有如下所示几个选项：

`-N job_name` 指定本次任务的名字，可以通过修改 `job_name` 的命名来让这次任务更容易识别和有意义，使用 `qstat` 命令查看正在运行的脚本时会显示这个名字。

`-o job.log, -e job.err` 分别是任务执行时的标准输出和标准错误输出的文件。

`-q` 执行计算的队列可以有 `low`、`middle`、`high`、`batch`、`fat`、`gpu` 选项，表示队列所属的优先级，在任务执行时可以执行指令 `qstat -q` 可以看到相关队列状况。

`-l walltime=1:00:00` 指定了本次任务所占用的机时上限，如 `walltime=02:00:00` 指定本次任务最长时间为两个小时，当超过后就会强制终止本次任务。**注意：这里设定的 `walltime` 并不是显示在 `Gridview` 中的计费机时 (`WallTime`)，如果在这个设定时间之前作业任务就结束了，那么多余的时间并不会计入计费机时。**

`-l nodes=5:ppn=24` 在并行作业中指定节点 (`nodes`) 数和每个节点的线程 (`ppn`) 数。**使用 `low`、`middle`、`high`、`batch` 队列的任务 `nodes` 数不能超过 26，每个 `node` 的线程数不能超过 28；使用 `fat` 队列的任务 `nodes` 数不能超过 2，每个 `node` 的线程数不能超过 144；使用 `gpu` 队列的 `nodes` 数不能超过 1，每个 `node` 的线程数不能超过 28（仅指 CPU 线程）。**

一般 PBS 文件中还可以加入 `-d` 参数来指定本次任务的工作目录，即作业读入输入文件和写输出文件的目录，`cd` 命令也可以进入某个目录来代替相关功能，如下图所示：

```
#!/bin/sh
#an example for Intel MPI job.
#DO NOT RUN THIS SCRIPT DIRECTLY,
#PLEASE RUN THIS SCRIPT WITH qsub: qsub intelmpi_job.pbs
#
#PBS -N mpi4py_manybody
#PBS -o job.log
#PBS -e job.err
#PBS -q high
#PBS -l nodes=5:ppn=24
#PBS -l walltime=1:00:00
cd /home/qkun/libin03/test_mpi/mpi4py
```