

河北大学高性能计算平台

使用说明

2022版

河北大学信息技术中心

2022-07

**目 录**

[1 高性能计算平台登陆 2](#_Toc25086)

[1) 登录方式一：Gridview 2](#_Toc8221)

[2) 登录方式二：ssh登录 5](#_Toc22096)

[1.2.1 SSH Secure Shell终端登录 5](#_Toc9395)

[1.2.2 Xshell登录 17](#_Toc24194)

[2 用户上机操作 22](#_Toc6750)

[1) 文件传输 22](#_Toc25186)

[2) 提交作业 24](#_Toc12599)

[2.2.1 用GridView web界面提交作业 24](#_Toc17162)

[2.2.2 用命令行方式提交作业 27](#_Toc31185)

[3) 作业管理 31](#_Toc6956)

[2.3.1 用GridView Web 界面管理作业 31](#_Toc1720)

[2.3.2 用命令行方式管理作业 34](#_Toc13096)

[4) 源码编译 37](#_Toc1743)

[2.4.1 c/fortran 编译器汇总选取 37](#_Toc20075)

[2.4.2 mpi 编译器汇总选取 38](#_Toc1936)

[2.4.3 环境变量的选取 38](#_Toc25894)

[超算中心集群系统数学库环境变量的选取 38](#_Toc18057)

[附录1 应用软件安装信息 40](#_Toc13408)

[附录2 队列信息 41](#_Toc29578)

[附录3 Material-Studio客户端配置 42](#_Toc8172)

[1) 添加Server Gateway 42](#_Toc4438)

[2) 查看Gateway的配置信息 47](#_Toc31955)

[3) 使用Gateway向服务端提交作业 48](#_Toc20702)

[附录4 常用软件pbs脚本模板 50](#_Toc20517)

[1) vasp-5.4.1-openmpi版本 50](#_Toc25419)

[2) vasp-5.2.12-openmpi版本 51](#_Toc11608)

[3) MaterialsStudio7.0 53](#_Toc24931)

[4) Gaussian09 55](#_Toc30878)

1. 高性能计算平台登陆

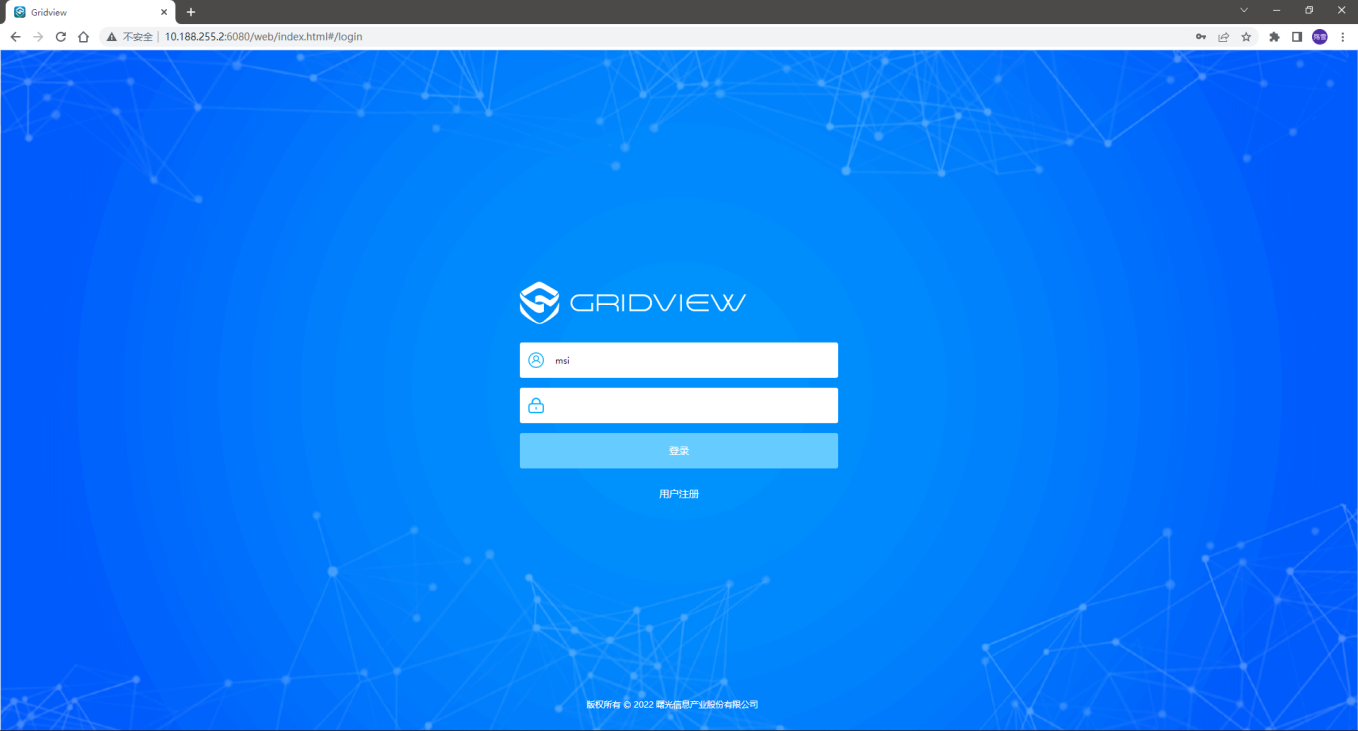
登录方式分为两种，一种是通过WEB浏览器登录高性能计算平台Gridview，另外一种方式是使用SSH远程登录高性能计算平台登录节点。

通过WEB浏览器登录Gridview后，可以查看平台资源情况（查看提交的作业数、可用核心数、授权可用的队列及队列核心数、授权可用的队列运行作业数）、作业提交、作业管理、资源申请和报表计费等主要功能。

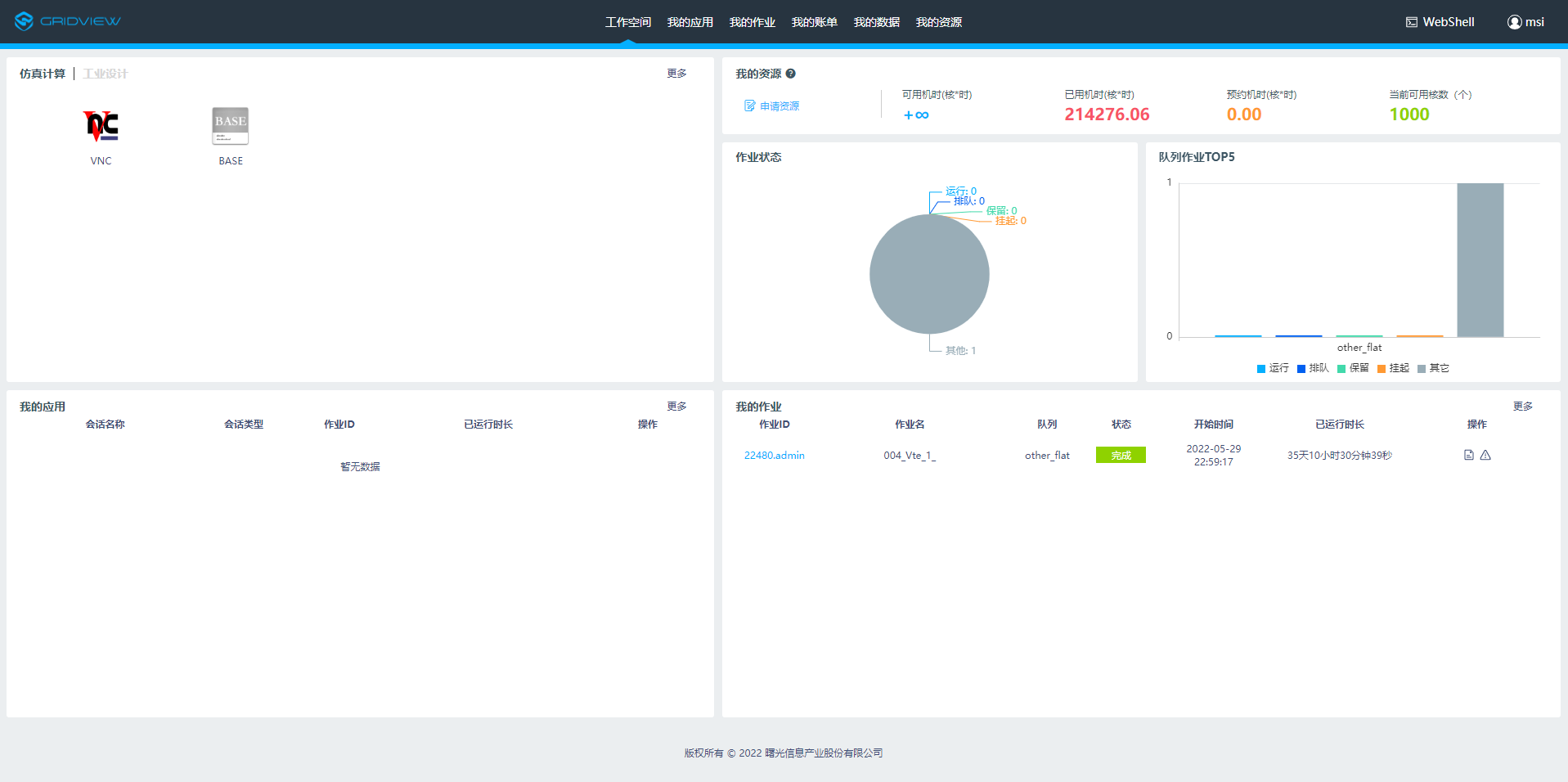
* 1. 登录方式一：Gridview

在firefox或chrome浏览器网址栏中输入[http://10.188.255.2:6080进入Gridview](http://hp.hbu.edu.cn进入Gridview)界面，如下图所示：s

1：输入用户名和密码之后进入个人用户界面目录。

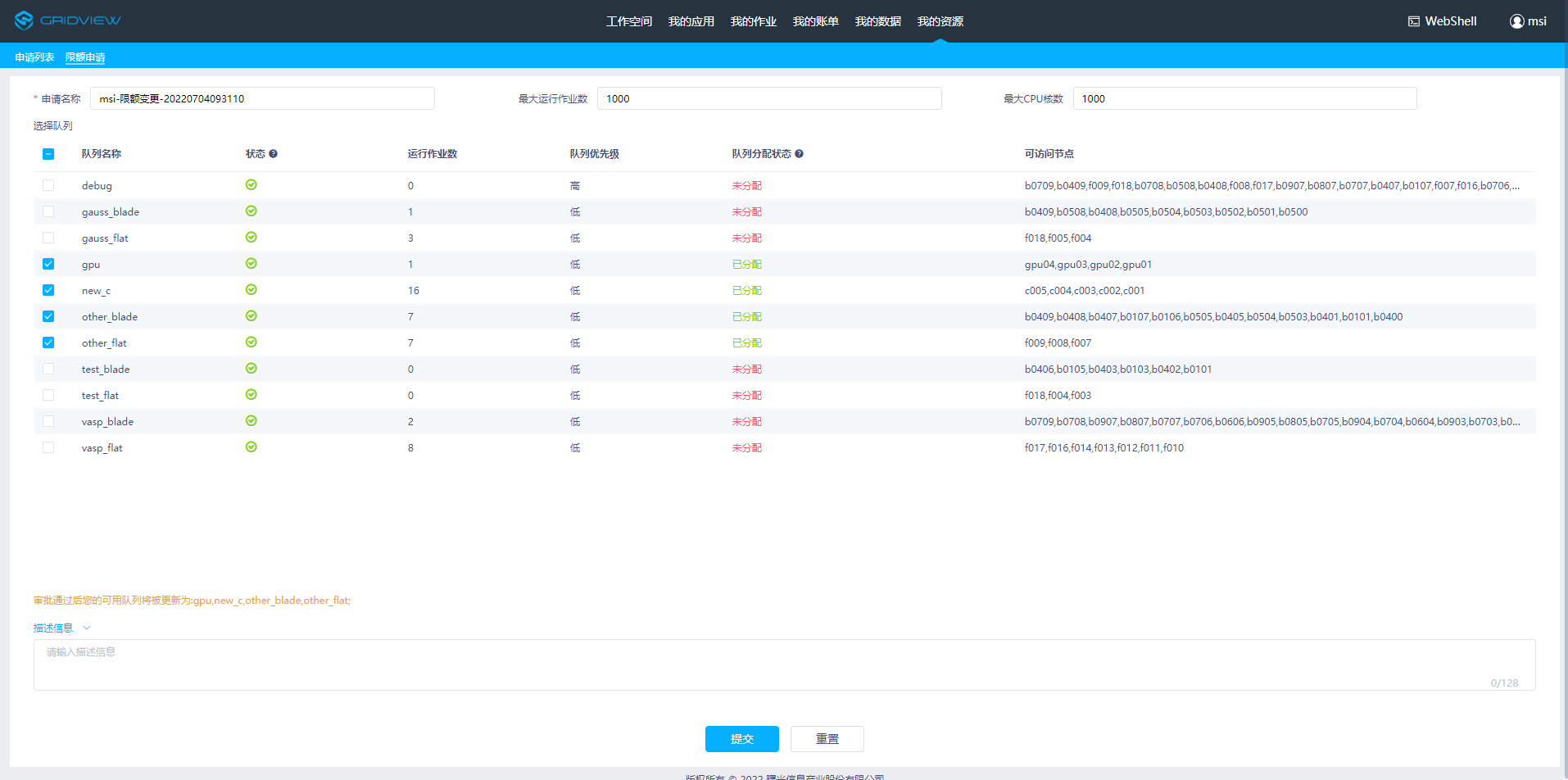


2：登录后的页面列出了常用的功能，如下：

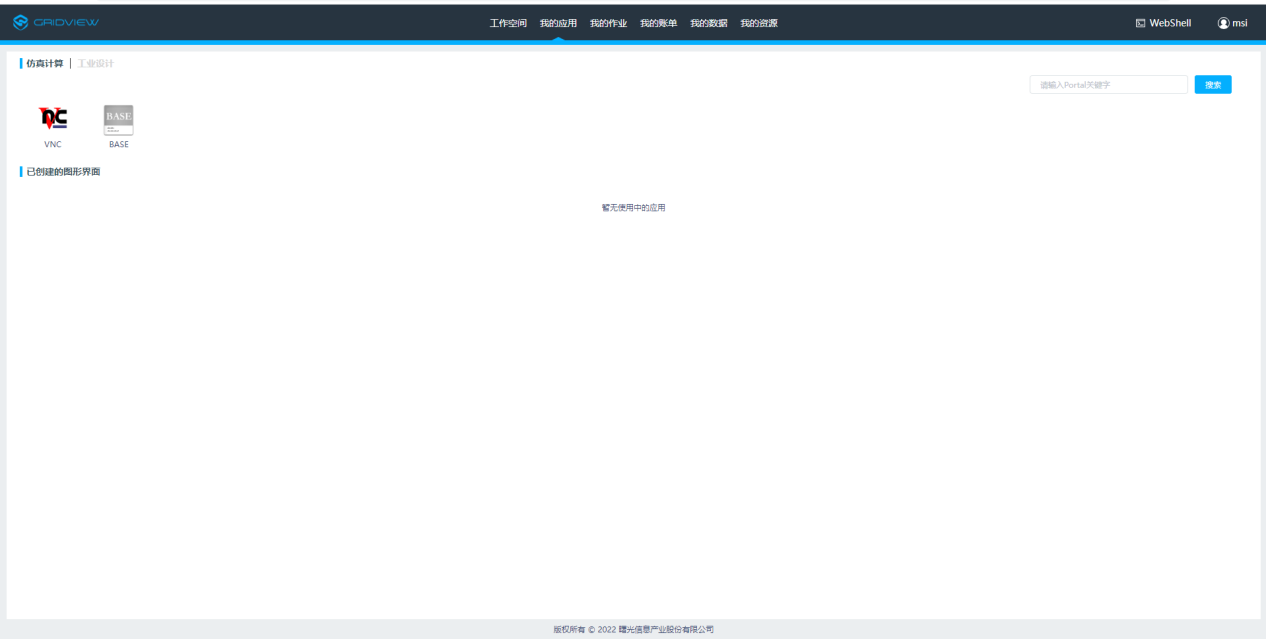


在主界面可以查看用户自己的资源，用户开启的应用，用户的作业状态。

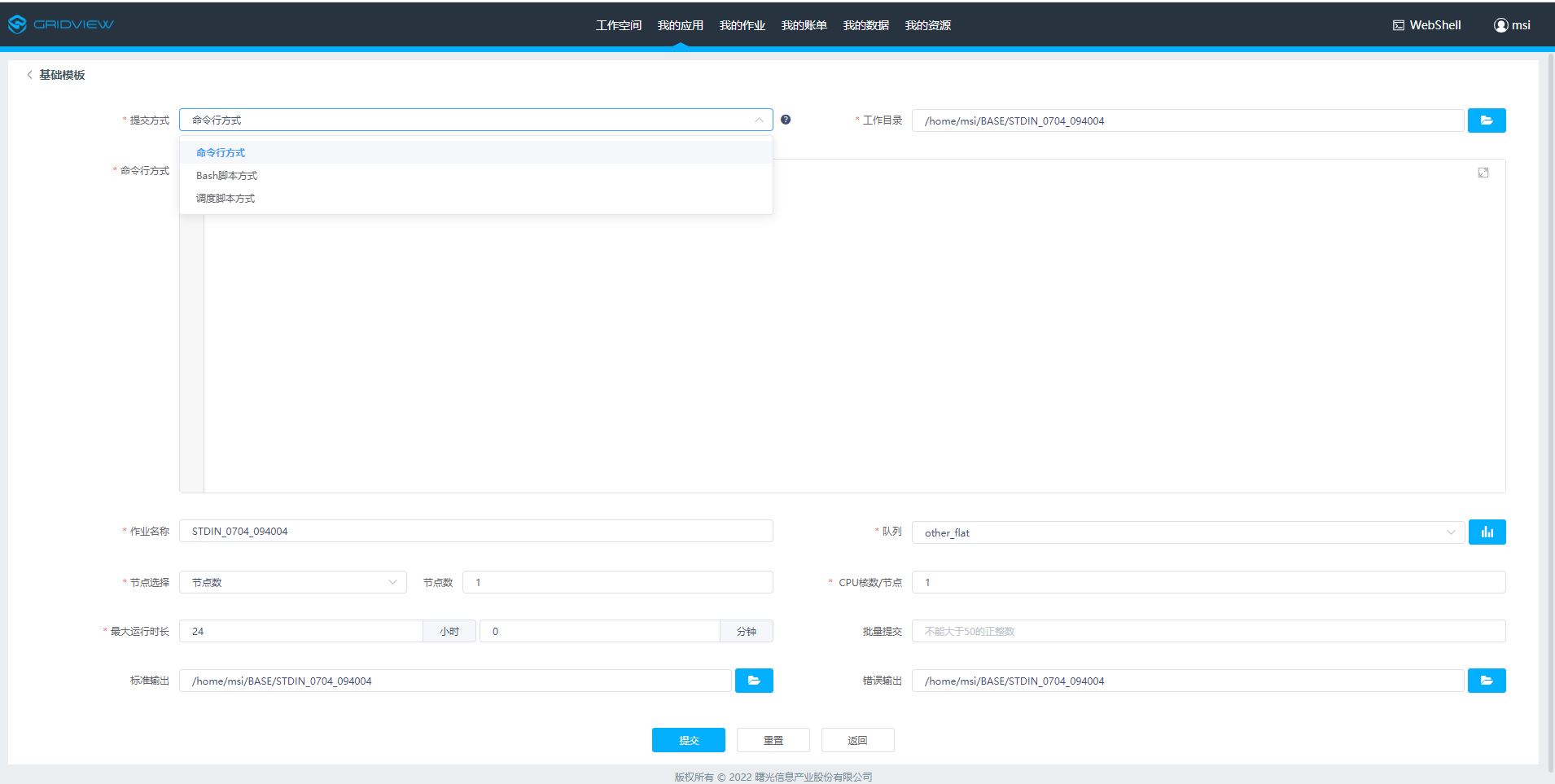
点击主界面上申请资源的队列按钮可以申请分配作业队列，此操作调教之后会将申请提交至管理员账户申请列表中，如您有需求可以联系信息中心的管理员老师进行核实并添加资源队列。



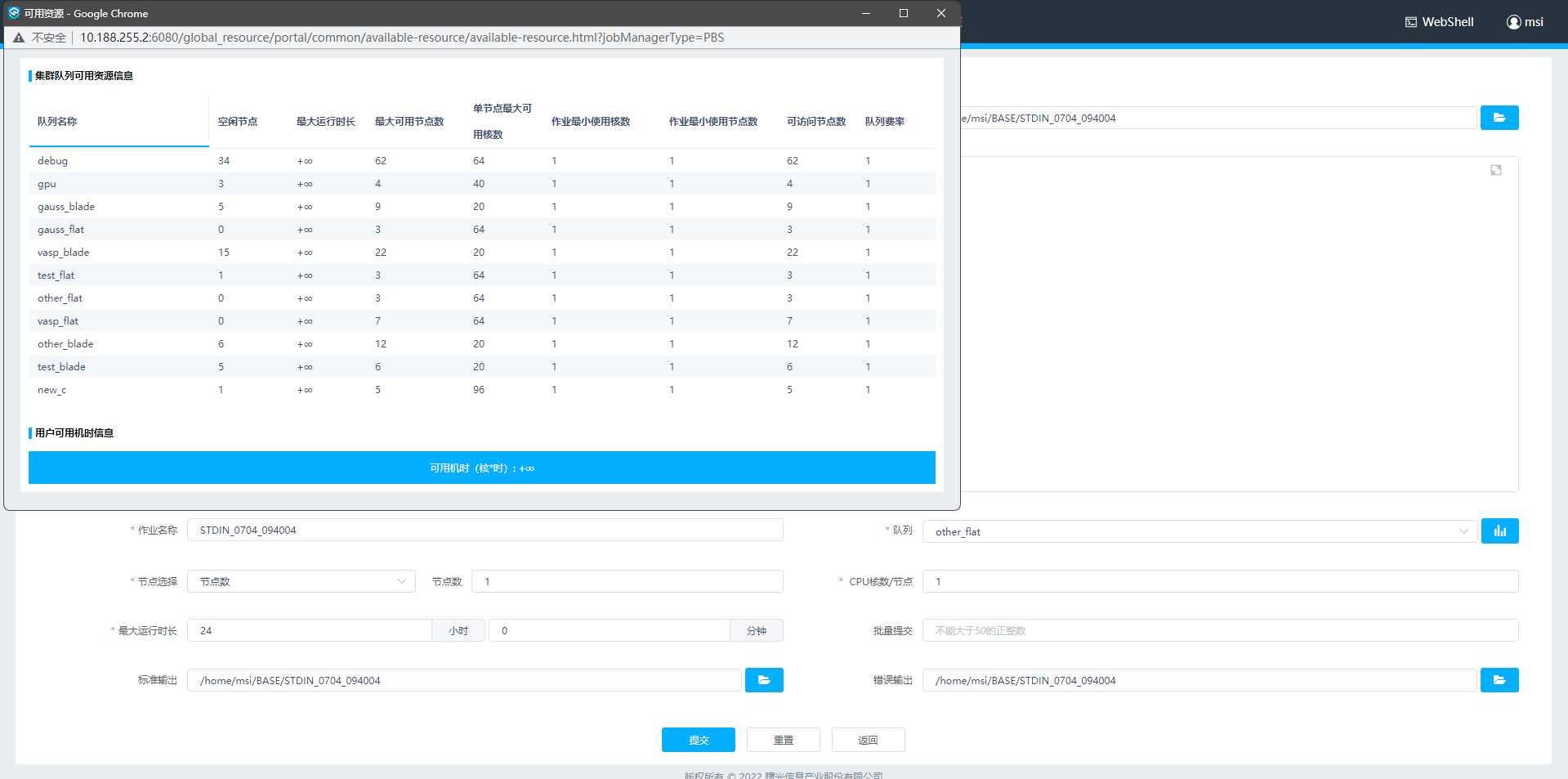
3：点击“我的应用”，进入应用列表，应用列表中会显示用户可使用的portal模板，在新的版本中我们已经将原本的mpi等模板集合到BASE模板中，用户提交可以使用通用的BASE模板。



点击BASE软件图标之后会进入基础模板，可以选择命令行、bash脚本、调度脚本的方式进行提交。



点击队列右侧的蓝色图标可以查看可用资源状态，可用状态内可以查看全部节点的状态，注意最大可用节点数、单节点最大可用核数、空闲节点数，之后撰写提交脚本需要使用到这些参数。



* 1. 登录方式二：ssh登录

高性能计算平台提供登陆域名http://10.188.255.2:6080，用户可以通过ssh、VNC等多种方式登录集群系统。

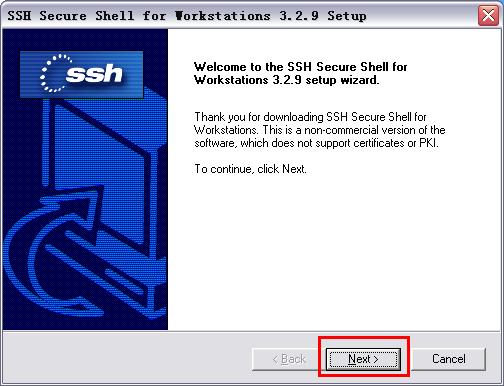
登录成功后默认位于高性能计算平台登录节点login2，禁止直接运行计算的软件和程序，所有计算任务和作业必须通过PBS脚本提交到计算节点运行。PBS脚本编写和提交的方法将在第2章节详细描述。

Windows用户可以用SSH Secure Shell Client，PuTTY，Xmanger等SSH客户端软件登录。本文以常用的SSH Secure Shell Client（集成了SFTP文件上传下载功能）和Xmanger为例分别介绍，请选择任意一种方式登录。

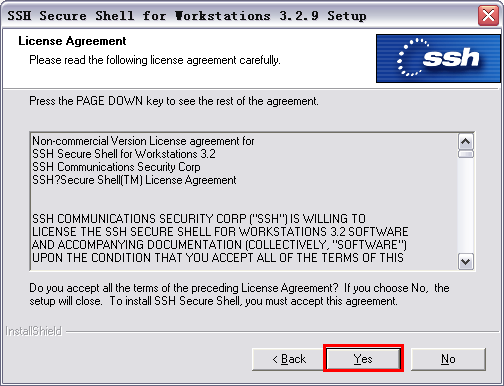
* + 1. SSH Secure Shell终端登录

SSH Secure Shell Client，下载地址：<http://dl.pconline.com.cn/download/398471-1.html>，

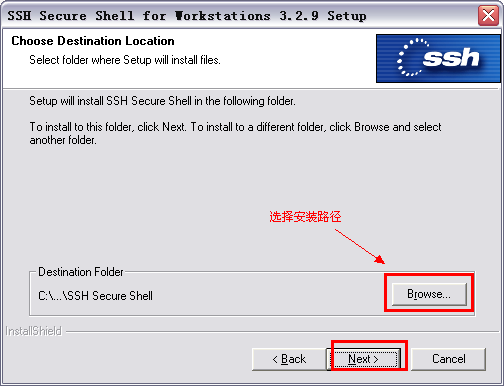
1：解压之后，双击exe文件，如下图所示，点击“Next”



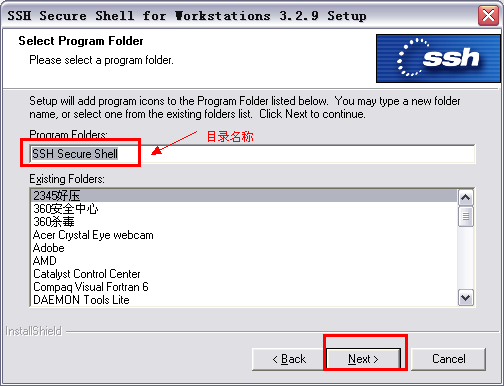
       2：同意安装协议，点击“Yes”按钮



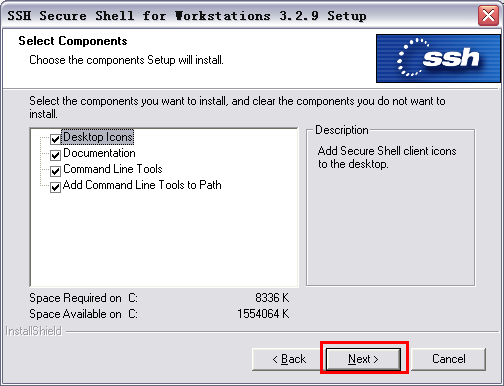
3：选择安装路径，然后点击下一步



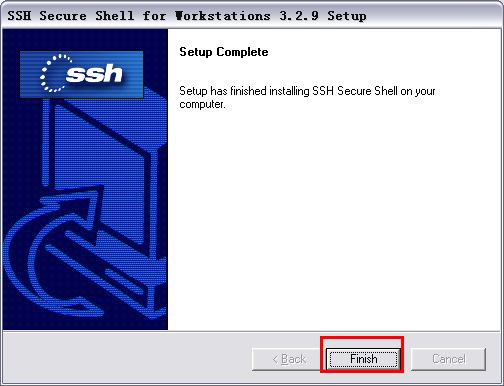
4：填写目录名称，然后点击“Next”



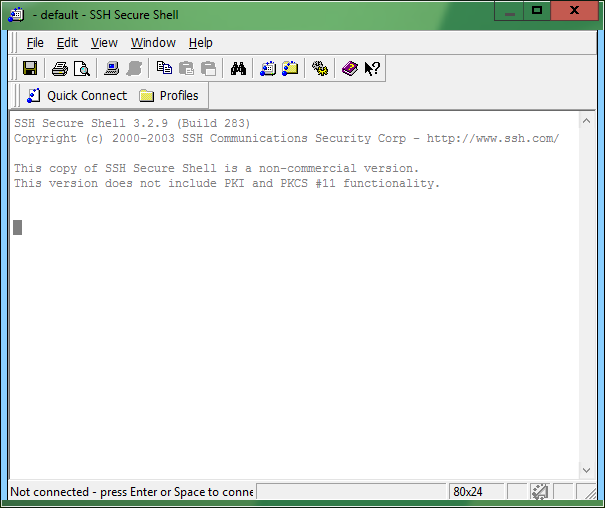
5：选择是否创建桌面图标、文档及其他，然后点击“Next”



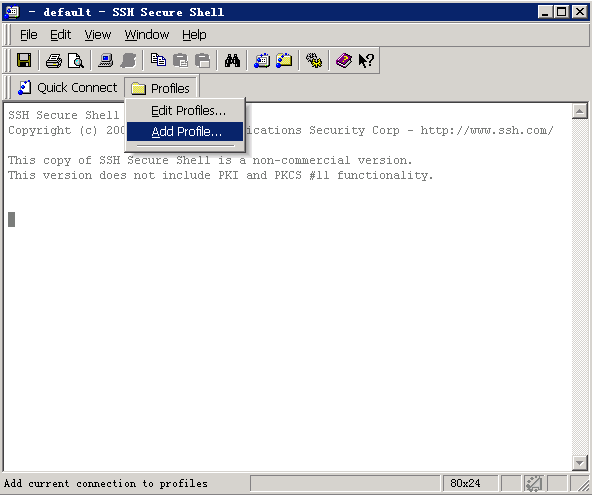
6：最后点“Finish”即可



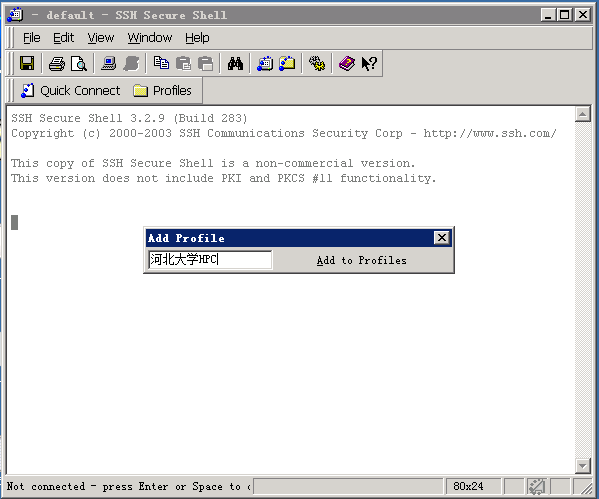
7：双击其客户端图标SSH Secure Shell Client，出现下图所示窗体



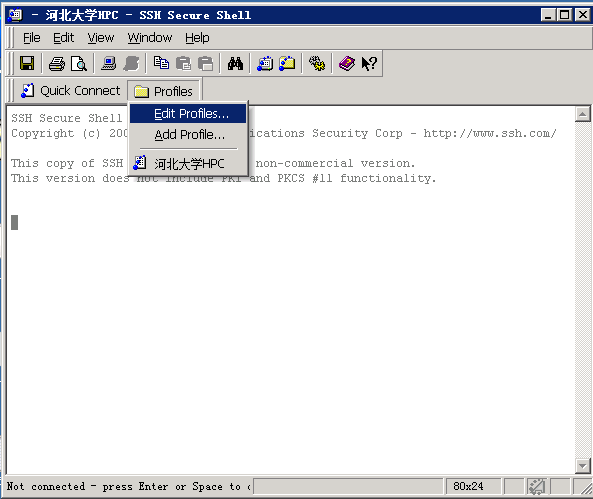
8：第一次选择新建一个Profile



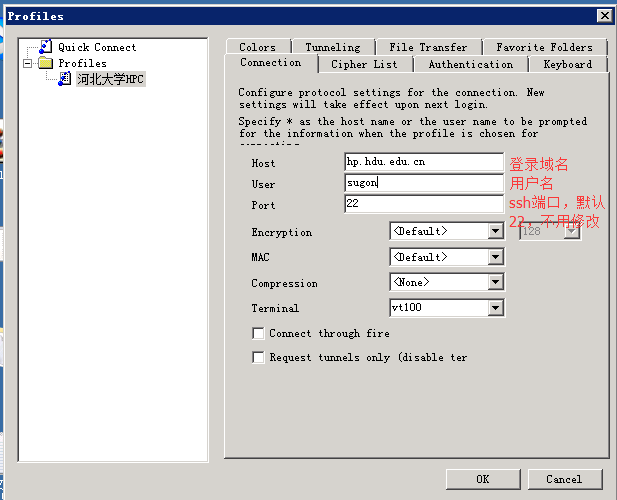
9：填写Profile的名称



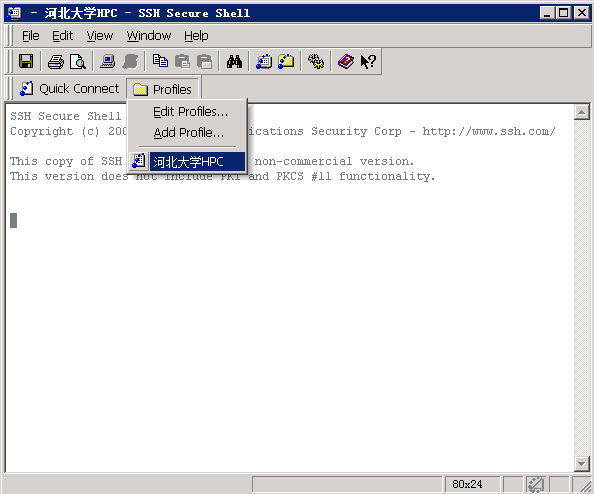
10：完善Profile的信息



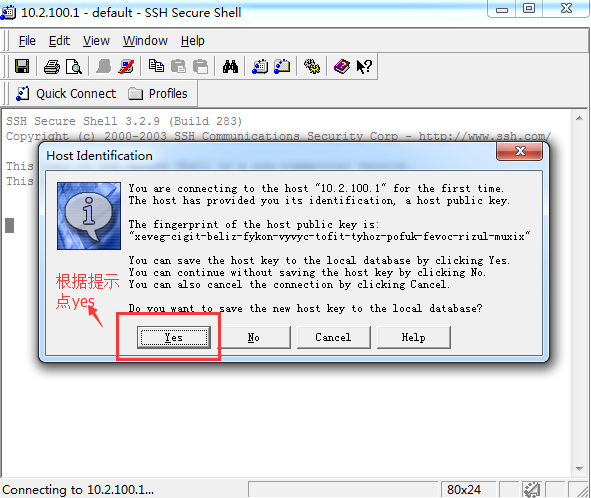
11：主要用填下远程主机的IP/USER(用户名)/PORT，在需要连接远程的主机的时候需要填写登录远程主机的PASSWORD，Host请填10.188.255.2，User请填在超算申请的用户名，Port默认22不需要修改。

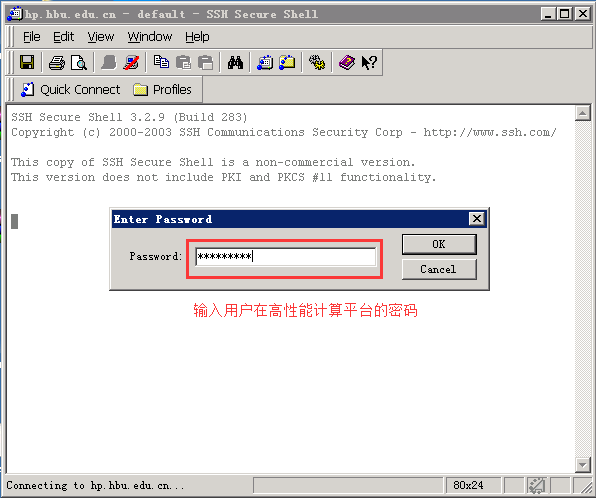


11：点击新建的Profile进行连接

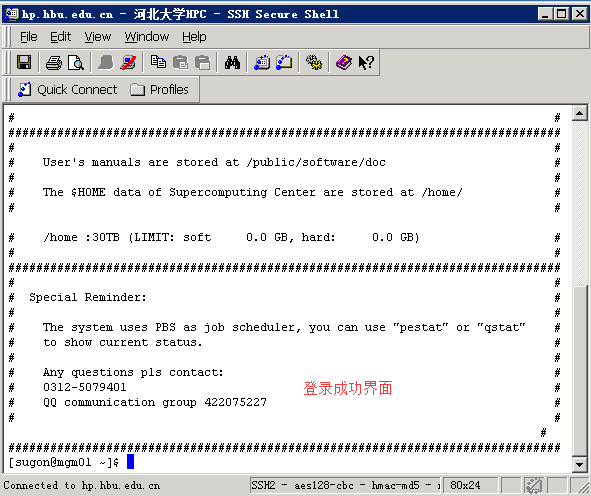


12：输入超算申请的用户登录密码





13：登录成功，可以操作高性能计算平台和本地计算机之间的交互操作



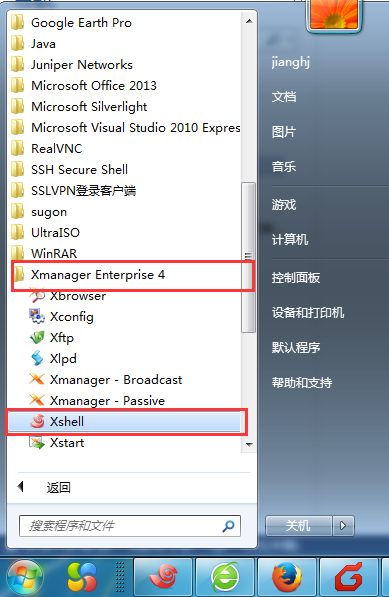
* + 1. Xshell登录

Xmanager 是全新标准的跨平台集成解决方案。它是一个一站式解决方案，这个软件包含有以下一些产品：Xmanager 3D(OpenGL)，[Xshell](http://baike.baidu.com/view/1036749.htm" \t "_blank)，[Xftp](http://baike.baidu.com/view/711531.htm" \t "_blank)，Xlpd和Xstart。

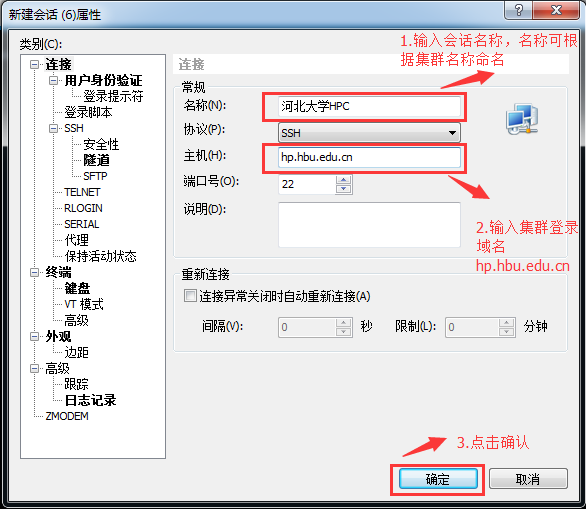
Xmanager为商业版软件，从互联网获取后，安装在笔记本或工作站终端，具体安装过程不再详细描述。

本文以Xmanager4的xshell为例演示，如何登录高性能计算平台，

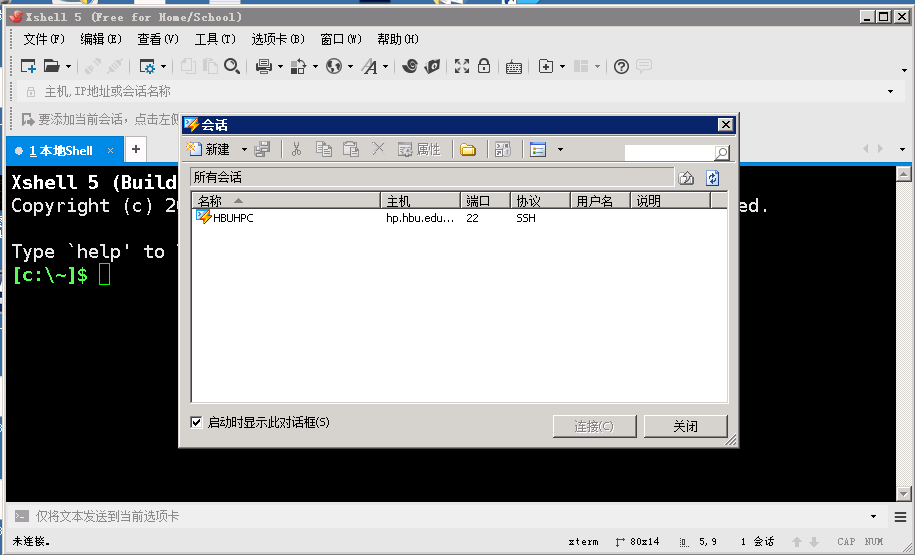
第一步，选择启动——程序——Xmanager4­——Xshell



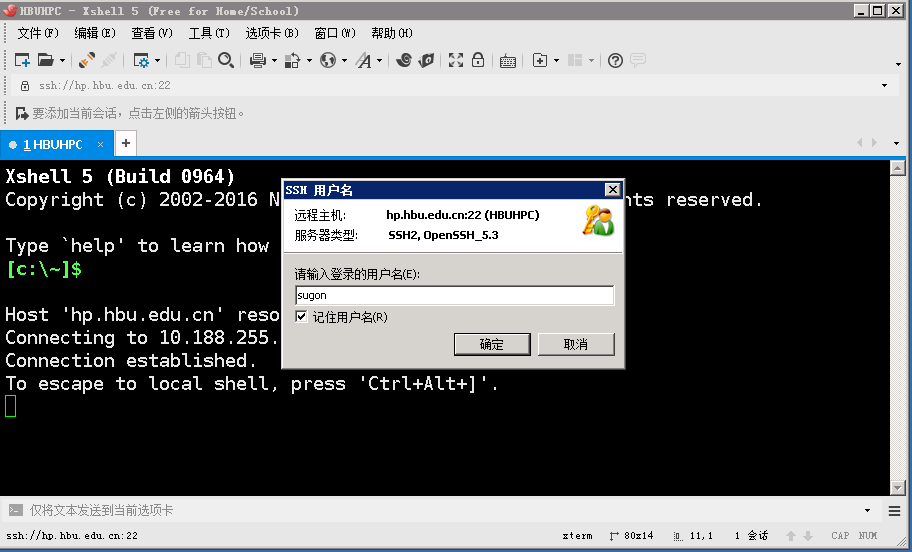
第二步，点击新建会话，依次填入集群名称、主机地址（10.188.255.2）



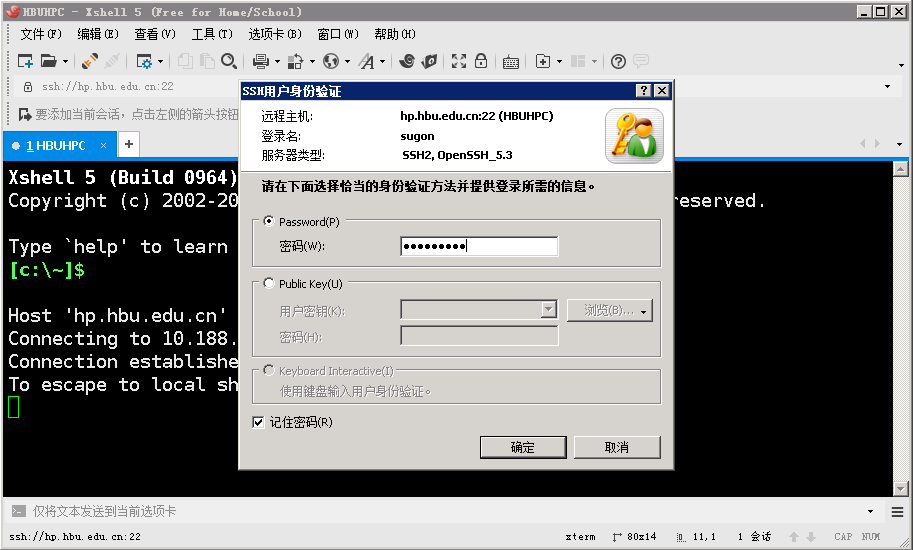
第三步，点击刚新建的会话



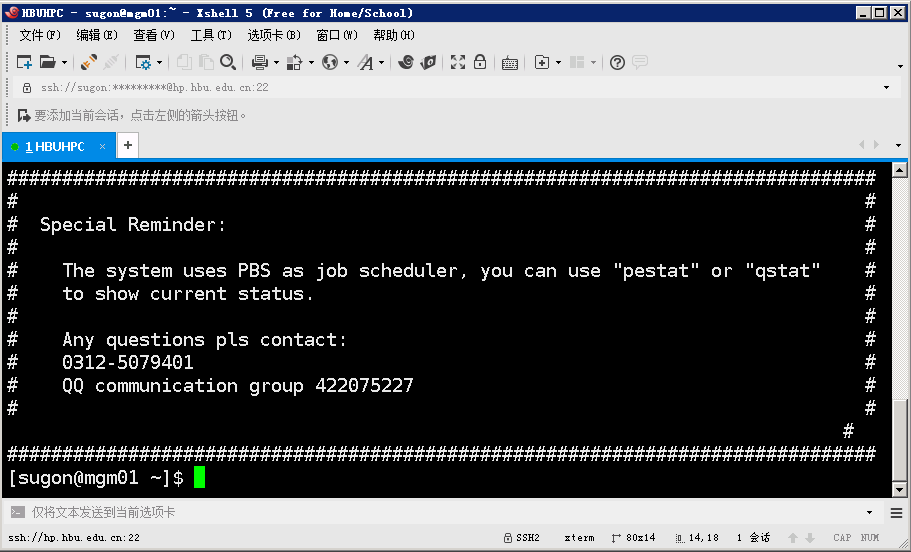
第四步，输入申请的用户名，为方便下次登录不再输入用户名，勾选记住用户名



第五步，输入申请的用户名密码，为方便下次登录不再输入密码，勾选记住密码



第六步，登录成功



1. 用户上机操作

在高性能计算平台，用户提交作业的基本步骤流程如下：

* 准备模型数据文件和作业脚本文件
* 用户登录河北大学高性能计算平台
* 传输文件到河北大学高性能计算平台
* 提交作业
* 作业管理
* 作业结果查看，将结果文件下载到本地硬盘

其中，准备模型数据文件和作业脚本文件，用户可以在本地完成。用户登录过程详见章节一，本文主要介绍其余几个步骤的进行方法。

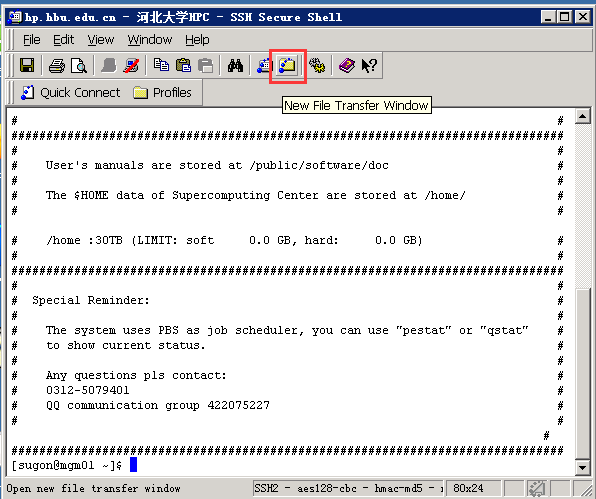
* 1. 文件传输

用户上传模型文件或者下载结果文件都需要进行文件传输。建议使用SFTP方式在高性能计算平台上传下载文件。

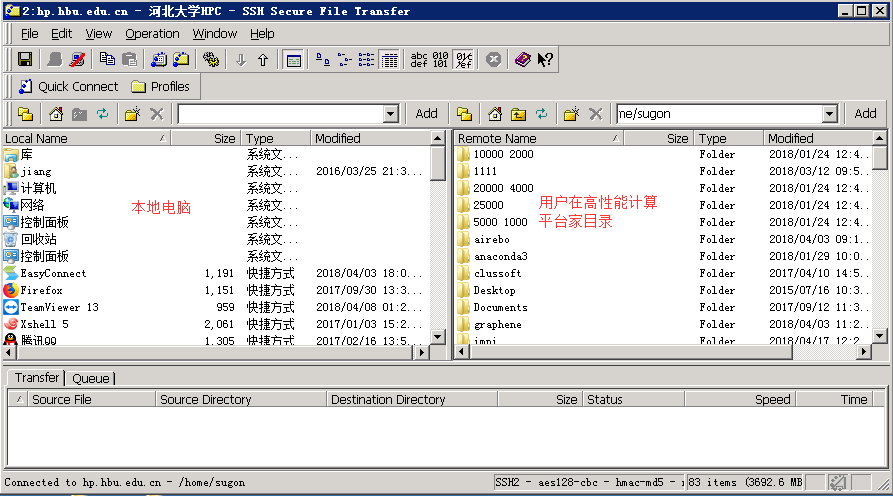
Windows用户可以用SSH Secure Shell Client，winscp等软件实现文件的上传下载。当需要下载或上传的文件数量较多时，建议用tar命令将文件进行打包后下载或上传，加快文件传输速度，减少文件传输的出错概率。

注意，因为windows操作系统和linux操作系统对于个别字符的处理方式不同，上传的文本文件在linux系统下，需要用dos2unix命令先处理一下，然后再继续使用，否则可能导致作业提交出错。

首先参考1.2.1章节登录高性能计算平台后，点击New File Transfer Window



连接成功后，可进行上传下载操作。



* 1. 提交作业

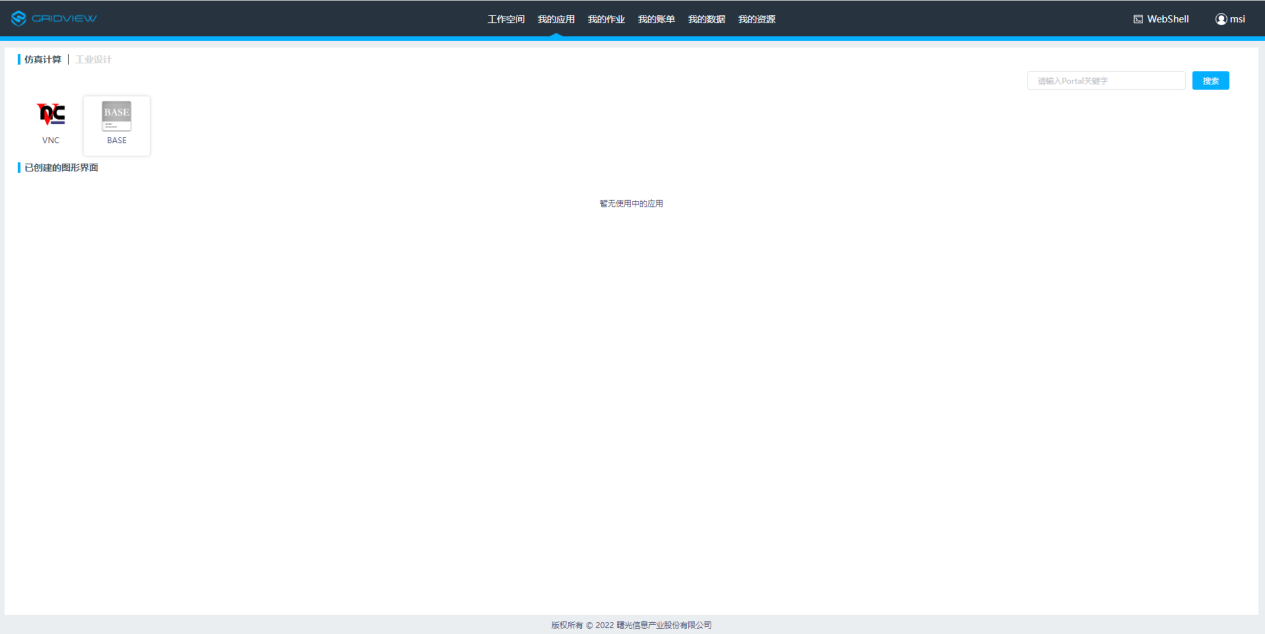
在高性能计算平台上提交作业，用户有两种方式可供选择。一种是通过GridView Web界面提交，一种是通过命令行的方式，用pbs命令提交。

**用户严禁使用任何前台或者后台方式直接运行程序，所有计算任务必须通过GridView web界面提交或命令行用PBS提交。**

具体提交作业方法如下所示。

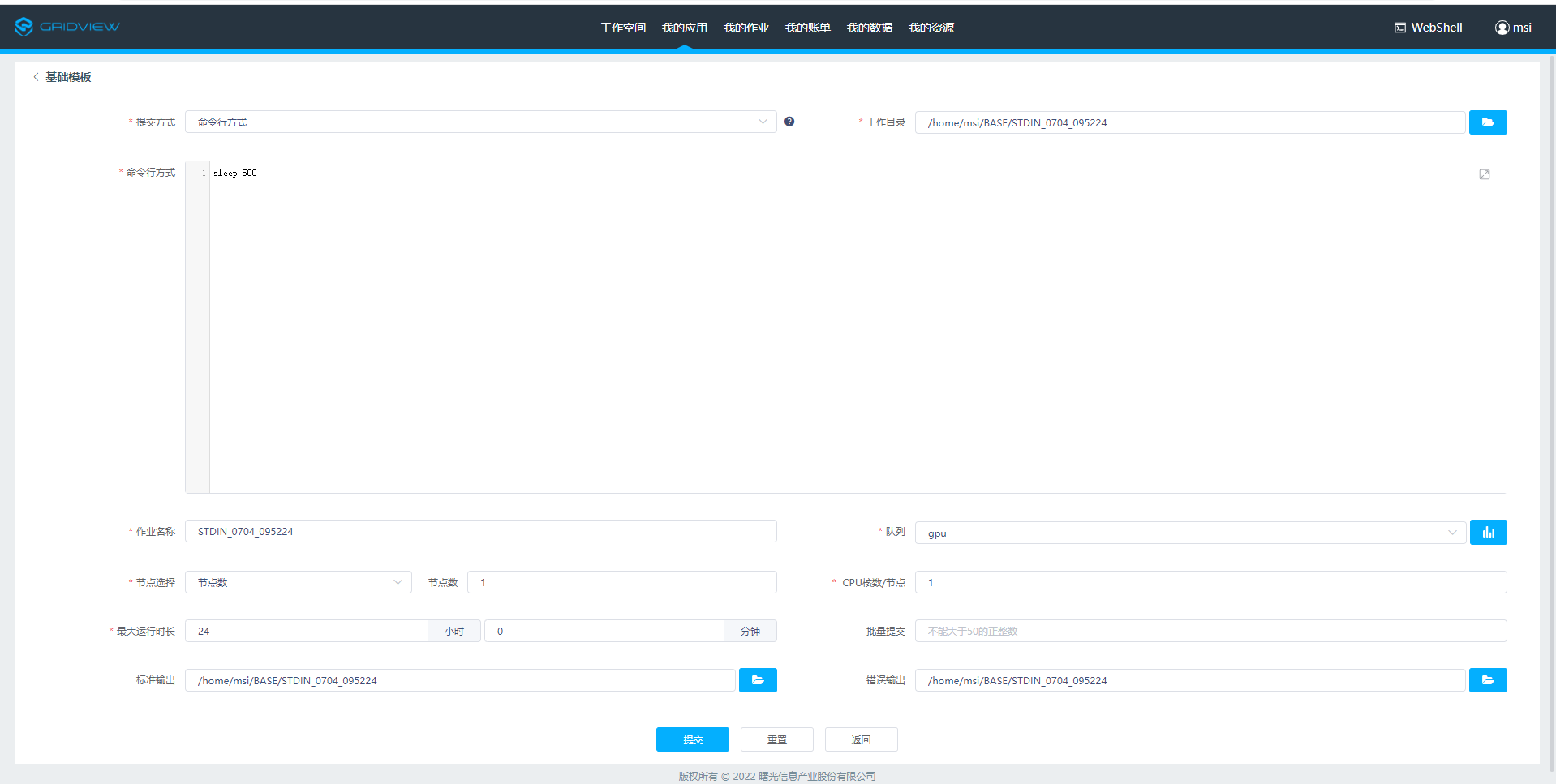
* + 1. 用GridView web界面提交作业

用户通过GridView web 界面提交作业，系统缺省可用portal有base基础模板和VNC启动，BASE模板可以提交并行作业和串行作业，如下图所示：



2.2.1.1BASE模板作业提交

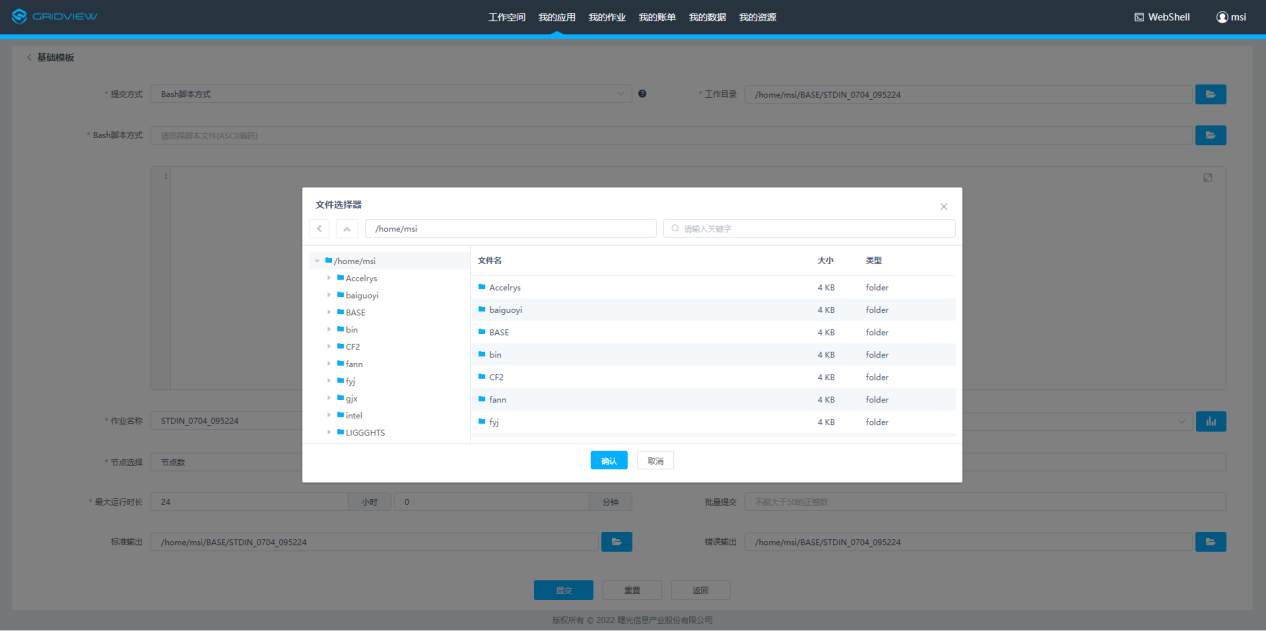
用户选择基础模板后，出现如下界面：



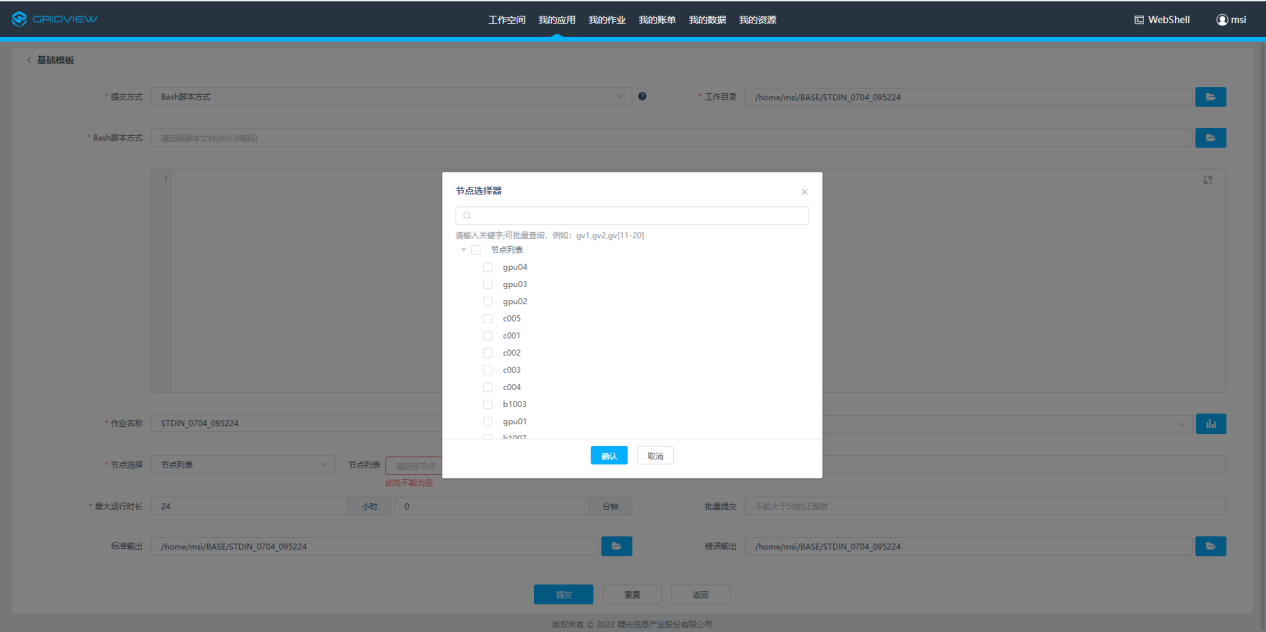
在该界面，标记为\*的为必须填写的栏目。

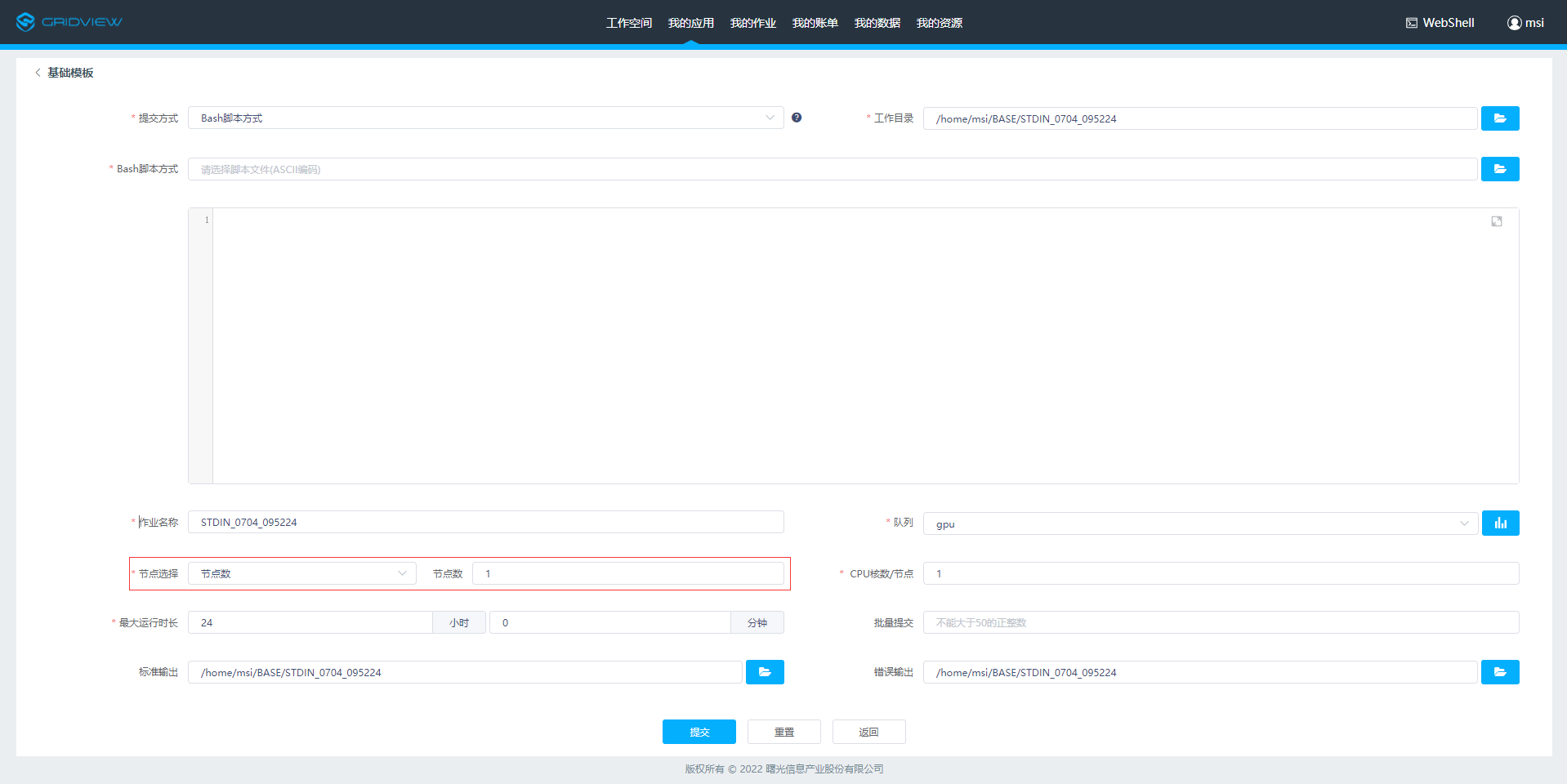
基础模板作业提交时，用户可以选择默认的集群名称，填写作业名称，或使用缺省名。用户可以根据自身作业的需要，选择不同的作业队列，关于队列的详细说明参见附录。工作路径默认为用户的家目录，用户可以根据需要选择。作业提交方式有命令行方式和脚本方式两种。用户仅能选择其中的一种。

如果选择用脚本方式提交作业，用户可以选择已经写好的脚本文件，如下图所示：

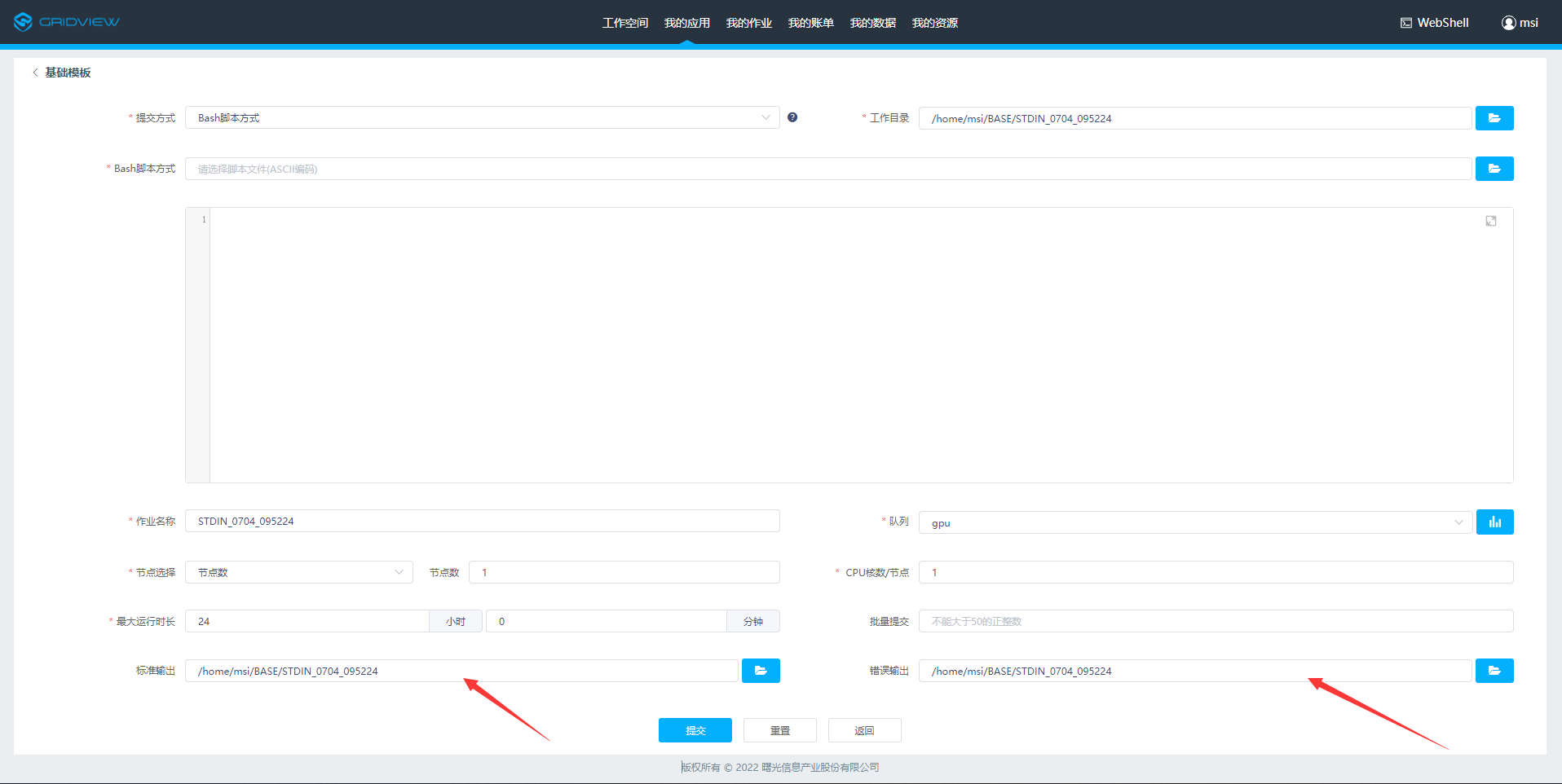


用户根据自己作业情况输入作业所需的执行时间，然后可以选择作业运行的节点数或节点列表。如下图所示：





节点数或节点列表选择完毕后，用户可以指定标准输出文件和错误输出文件名字和路径，也可以用缺省设置。



* + 1. 用命令行方式提交作业

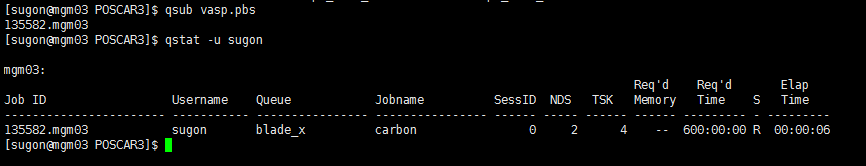
用户可以通过ssh客户端软件直接登录到河北大学高性能计算平台上，用命令行的方式提交作业。请注意，用户可以在登录节点查看文件，编辑文件，查看作业，查看资源使用情况等。不允许在登陆节点上运行计算程序。

在登录节点上提交作业只允许通过pbs作业调度系统提交作业。Pbs提交作业是通过qsub命令来执行，qsub命令通过脚本文件提交作业到作业管理系统，具体格式如下：

qsub <PBS作业脚本>

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



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已经写好的PBS脚本然后按照需求来进行修改

PBS作业脚本本质上是一个shell脚本，注释行以“#”开头，pbs运行参数以“#PBS”开头。PBS作业脚本里可以直接调用shell命令和系统命令。脚本里运行参数说明如下表所示：

|  |  |
| --- | --- |
| 运 行 参 数 | 说 明 |
| -a <作业开始运行的时间 | 向PBS系统指定作业运行的开始时间。  作业运行时间格式为： [[[[CC]YY]MM]DD]hhmm[.SS] |
| -A <用户名> | 使用不同的用户来提交作业，缺省使用当前用户名 |
| -o <标准输出文件的路径  -e <标准错误输出的路径 | 该参数指定标准错误输出的位置，缺省的情况下，PBS系统把标准输出和标准错误输出放在用户qsub命令提交作业的目录下。  标准错误输出：<作业名>.o<作业号>  标准错误输出：<作业名>.e<作业号>  路径使用如下格式标准： [<节点名>:]<路径名> |
| -N <作业名> | 指定提交的作业名 |
| -q <目标队列> | 指定作业提交的目标队列，其中目标队列可以是目标队列、目标节点名或者是目标节点上的队列。如果目标队列是一个路由队列，那么服务器可能把作业路由到新的队列中。如果该参数没有指定，命令qsub会把作业脚本提交到缺省的队列中。 |
| -l <申请资源列表> | 该参数指定作业脚本申请的PBS系统资源列表。  申请资源列表使用如下格式：  <资源名>[=[<数量>]][,资源名[=[<数量>]]， …..]  例如作业希望申请在双路节点上申请5个CPU资源的情况，  则可以在脚本中如下：  #PBS –l nodes=2:ppn=2+1:ppn=1 |
| 登陆SHELL继承来的变量 | 包括$HOME，$LANG，$LOGNAME，$PATH，$MAIL，$SHELL和$TZ。 |
| $PBS\_O\_HOST | qsub提交的节点名称 |
| $PBS\_O\_QUEUE | qsub提交的作业的最初队列名称 |
| $PBS\_O\_WORKDIR | qsub提交的作业的绝对路径 |
| $PBS\_JOBID | 作业被PBS系统指定的作业号 |
| $PBS\_JOBNAME | 用户指定的作业名，可以在作业提交的时候用qsub –N <作业名>指定，或者在PBS脚本中加入#PBS –N <作业名>。 |
| $PBS\_NODEFILE | PBS系统指定的作业运行的节点名。该变量在并行机和机群中使用。当在PBS脚本中用#PBS –l nodes=2:ppn=2指定程序运行的节点数时，可以使用$PBS\_NODEFILE在脚本中引用PBS系统指定的作业运行的节点名。比如：  #PBS –l nodes=2:ppn=2  mpirun –np 4 –machinefile $PBS\_NODEFILE <程序名> |
| $PBS\_QUEUE | PBS脚本在执行时的队列名 |

在集群计算系统上提交的作业通常分为如下几类：

* 普通的串行程序，仅使用一个计算核心即可。
* 同一节点内运行的OpenMP或基于threads的共享内存程序，仅使用一个节点内的多个核心。
* 利用消息传递方式的跨节点的MPI并行程序。
* 复杂的OpenMP+MPI混合并行程序。

注意：以下作业脚本示例均以debug队列为例，请修改成您被授权可以使用的队列。

2.2.2.1普通的串行程序作业脚本示例：

|  |  |
| --- | --- |
| 脚本 | 脚本说明 |
| #PBS -N test | 指定作业名：test |
| #PBS -l nodes=1:ppn=1 | 指定作业资源：1个节点，1个核心 |
| #PBS -l walltime=12:00:00 | 指定作业运行时间 12天 |
| #PBS -q debug | 指定作业运行队列 blade\_s1 |
| cd $HOME/test/ | 进入工作目录 $HOME/test |
| ./a.out > $HOME/result/a.result | 执行命令，并将结果重定向到 ./a.out > $HOME/result/a.result 文件中。 |

上面为一个普通的串行作业脚本示例，用户可以通过qsub命令，加上该脚本的文件名，就可提交作业。脚本中给指定了作业名称，作业所需资源，作业的运行时间，作业运行所用队列，以及作业执行的目录。用户把作业的可执行文件和目录更换到用户的自己的内容就可成功提交用户自己的作业。

2.2.2.2共享内存并行作业

|  |  |
| --- | --- |
| 脚本 | 脚本说明 |
| #PBS -N test\_OpenMP | 指定作业名：test\_OpenMP |
| #PBS -l nodes=1:ppn=10 | 指定作业资源：1个节点，10个核心 |
| #PBS -l walltime=12:00:00 | 指定作业运行时间 12天 |
| #PBS -q debug | 指定作业运行队列 blade\_s1 |
| export OMP\_NUM\_THREADS=10 | 设置OpenMP线程数环境变量 |
| cd $HOME/test\_OpenMP/ | 进入工作目录 $HOME/test\_OpenMP |
| ./a.out > $HOME/result/a.result.OpenMP | 执行命令，并将结果重定向到 ./a.out > $HOME/result/a.result.OpenMP 文件中。 |

该类作业包括OpenMP并行方式的作业，也包括不使用OpenMP而是通过POSIX等系统底层所编写的多线程程序。

**用户请注意，这里的用户申请的节点数，核心数，需要同OMP\_NUM\_THREADS一致，且该数值不应该超出队列中单节点的物理核心数。同时，还要注意用户可执行程序的输入文件如需要设定OpenMP的核心数，也要同上面的参数设置一致。**

2.2.2.3 MPI并行作业

|  |  |
| --- | --- |
| 脚本 | 脚本说明 |
| #PBS -N test\_MPI | 指定作业名：test\_MPI |
| #PBS -l nodes=2:ppn=20 | 指定作业资源：2个节点，40个核心 |
| #PBS -l walltime=12:00:00 | 指定作业运行时间 12天 |
| #PBS -q debug | 指定作业运行队列 blade\_s1 |
| echo This jobs is $PBS\_JOBID@$PBS\_QUEUE | 打印作业名字和作业所属队列信息 |
| cd $PBS\_O\_WORKDIR | 进入工作目录 $PBS\_O\_WORKDIR |
| NP=`cat $PBS\_NODEFILE|wc -l` | 获取作业资源 |
| mpirun -np $NP -machinefile $PBS\_NODEFILE ./vasp | 运行Vasp可执行文件，指定40个线程，以及所用的节点名 |

该类作业为MPI并行方式的作业，**请用户注意设定的核心数数值不应该超出队列中单节点的物理核心数**。

2.2.2.4 OpenMP+MPI混合并行作业

|  |  |
| --- | --- |
| 脚本 | 脚本说明 |
| #PBS -N test\_OMP\_MPI | 指定作业名：test\_OMP\_MPI |
| #PBS -l nodes=2:ppn=20 | 指定作业资源：2个节点，40个核心 |
| #PBS -l walltime=12:00:00 | 指定作业运行时间 12天 |
| #PBS -q debug | 指定作业运行队列 blade\_s1 |
| echo This jobs is $PBS\_JOBID@$PBS\_QUEUE | 打印作业名字和作业所属队列信息 |
| cd $PBS\_O\_WORKDIR | 进入工作目录 $PBS\_O\_WORKDIR |
| NP=`cat $PBS\_NODEFILE|wc -l` | 获取作业资源 |
| mpirun -np $NP -machinefile $PBS\_NODEFILE **--mca btl self,openib,sm** ./test\_OMP\_MPI | 运行test\_OMP\_MPI可执行文件，指定40个线程，以及所用的节点名，并指定共享内存，和用InfiniBand网络计算 |

该类作业为OpenMP+MPI混合并行方式的作业，**请用户注意用户设定的核心数数值不应该超出队列中单节点的物理核心数**。

2.2.2.5深度学习作业提交

超算中心的集群系统除cpu服务器外，还包括gpu节点。对于gpu节点上的作业提交，需要计算程序需要支持gpu。开通gpu节点使用权限，请联系信息技术中心。

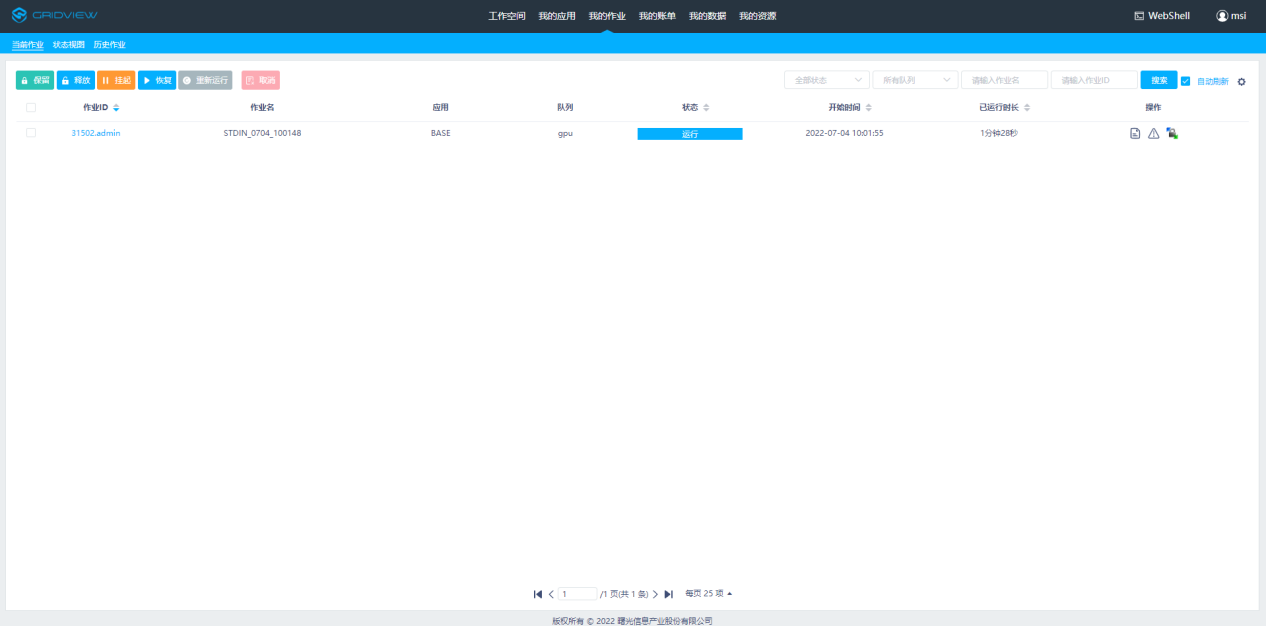
* 1. 作业管理

在超算中心的集群系统上进行作业管理，用户有两种方式可供选择。一种是通过GridView Web界面，一种是通过命令行的方式，用pbs命令管理查询。

* + 1. 用GridView Web 界面管理作业

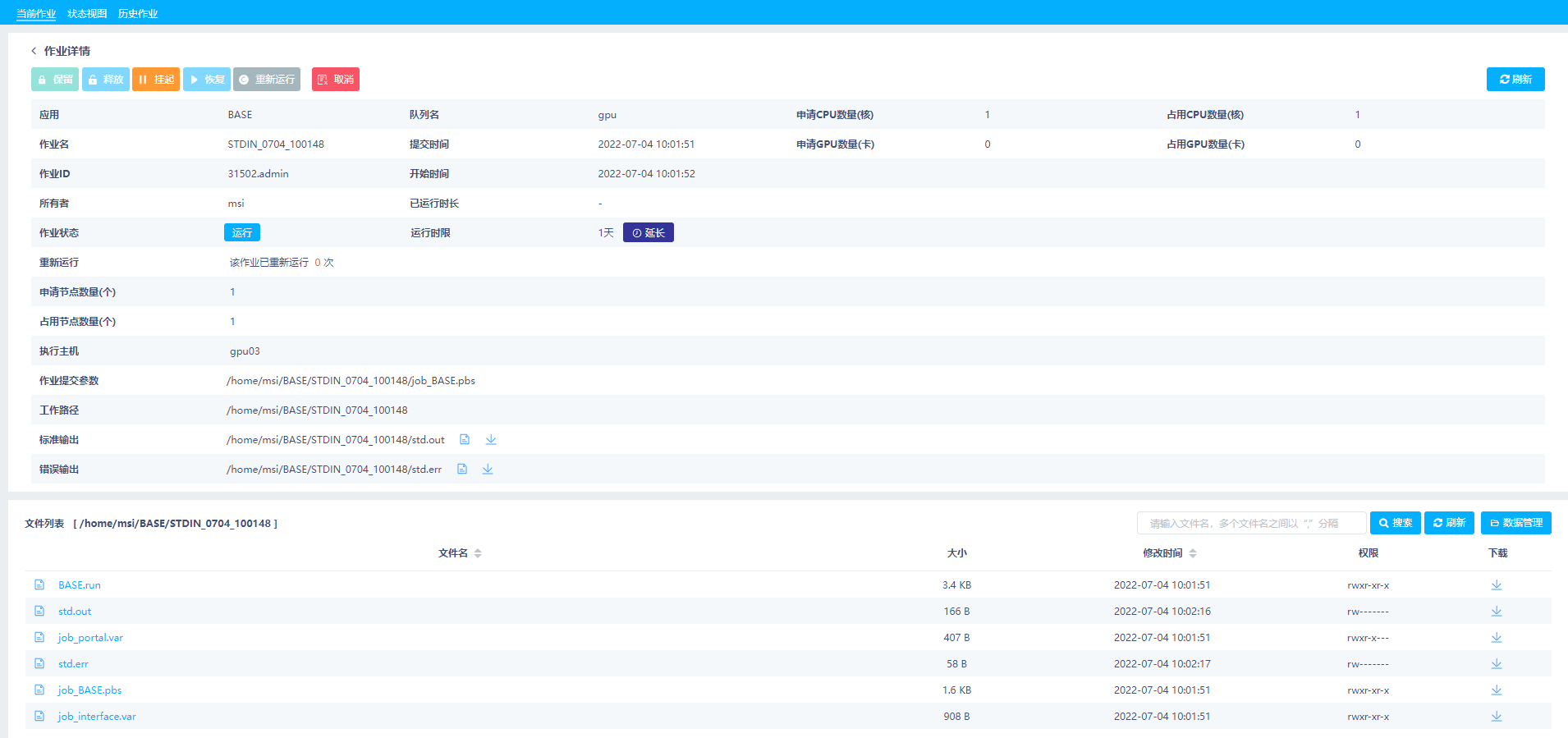
2.3.1.1 用户作业状态查询

提交过作业后，可以点击作业列表菜单查询作业状态，如下图所示：



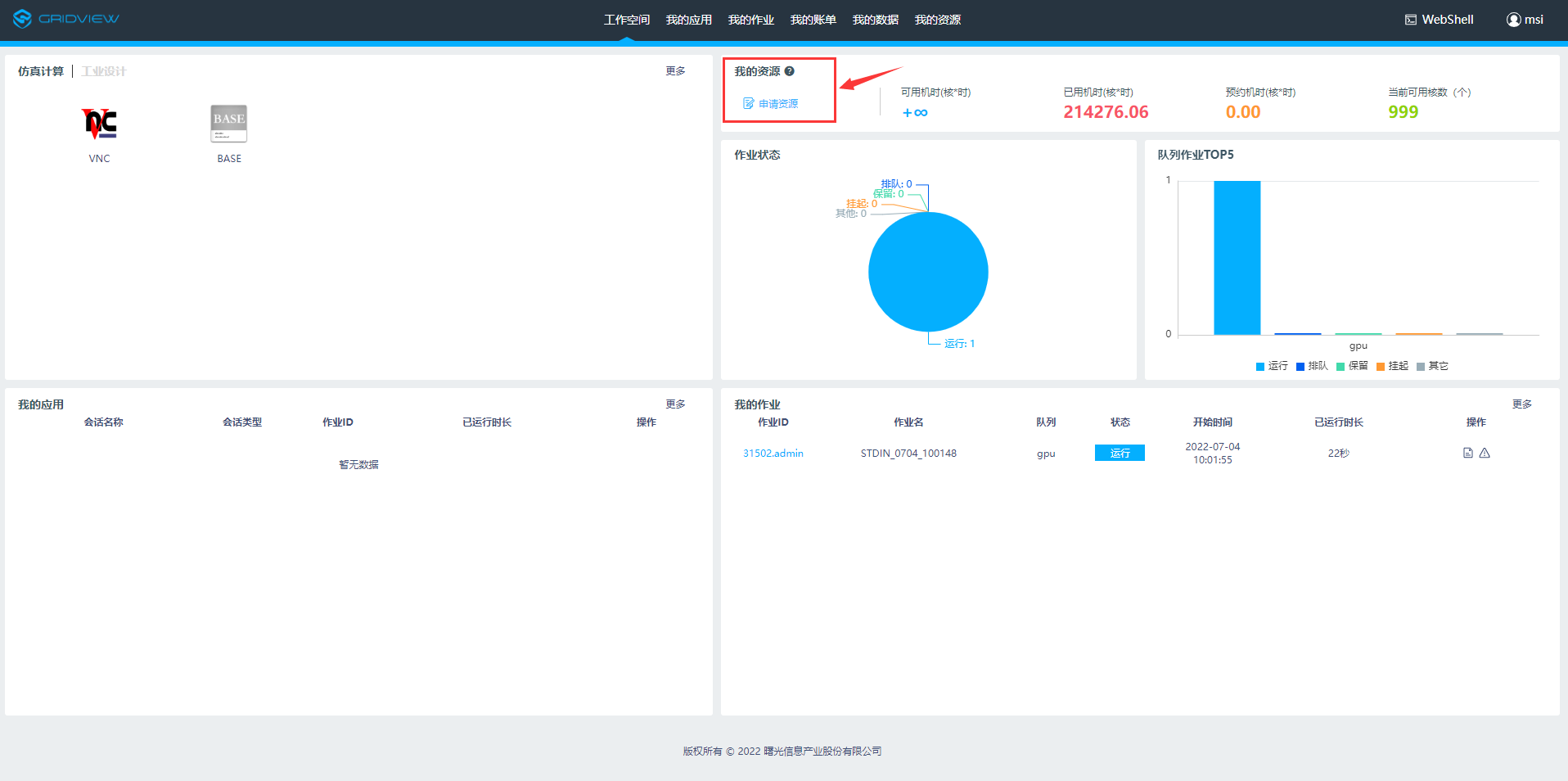
2.3.1.2 用户作业管理

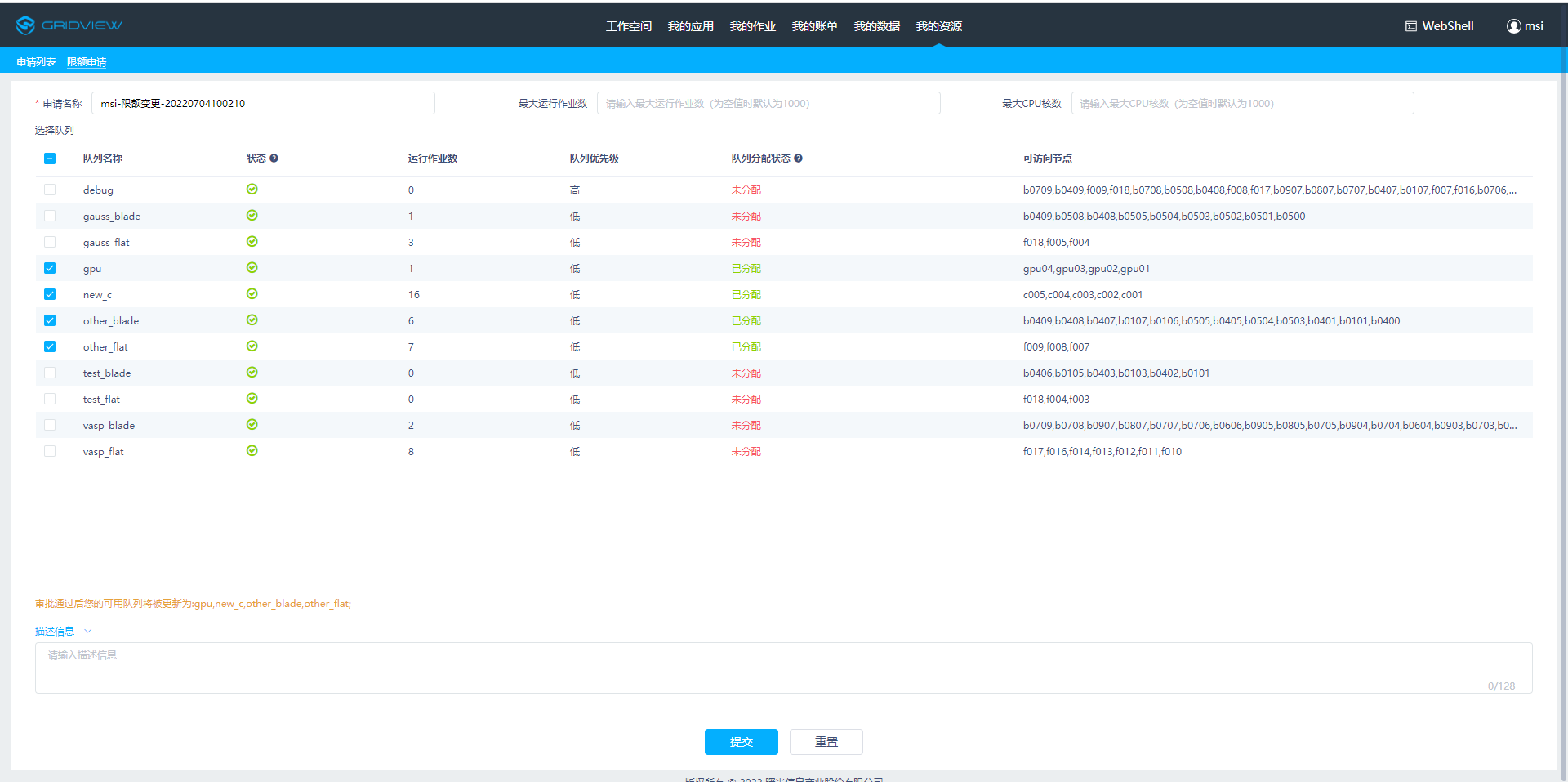
用户可以对列表中队列进行操作。**注意用户可以看到其他用户作业的运行状态，但用户并不能对其他用户的作业进行操作，仅能对于自己的作业进行操作。**操作方式是点选左上角的快捷按钮，可以删除作业，保留作业，挂起作业，恢复作业，释放作业，以及重新运行作业等。如下图所示：



2.3.1.3 用户可用资源申请

用户还可以对自身可用资源进行申请(最大作业数，最大核心数灯)，需要点击左侧的资源申请菜单项：

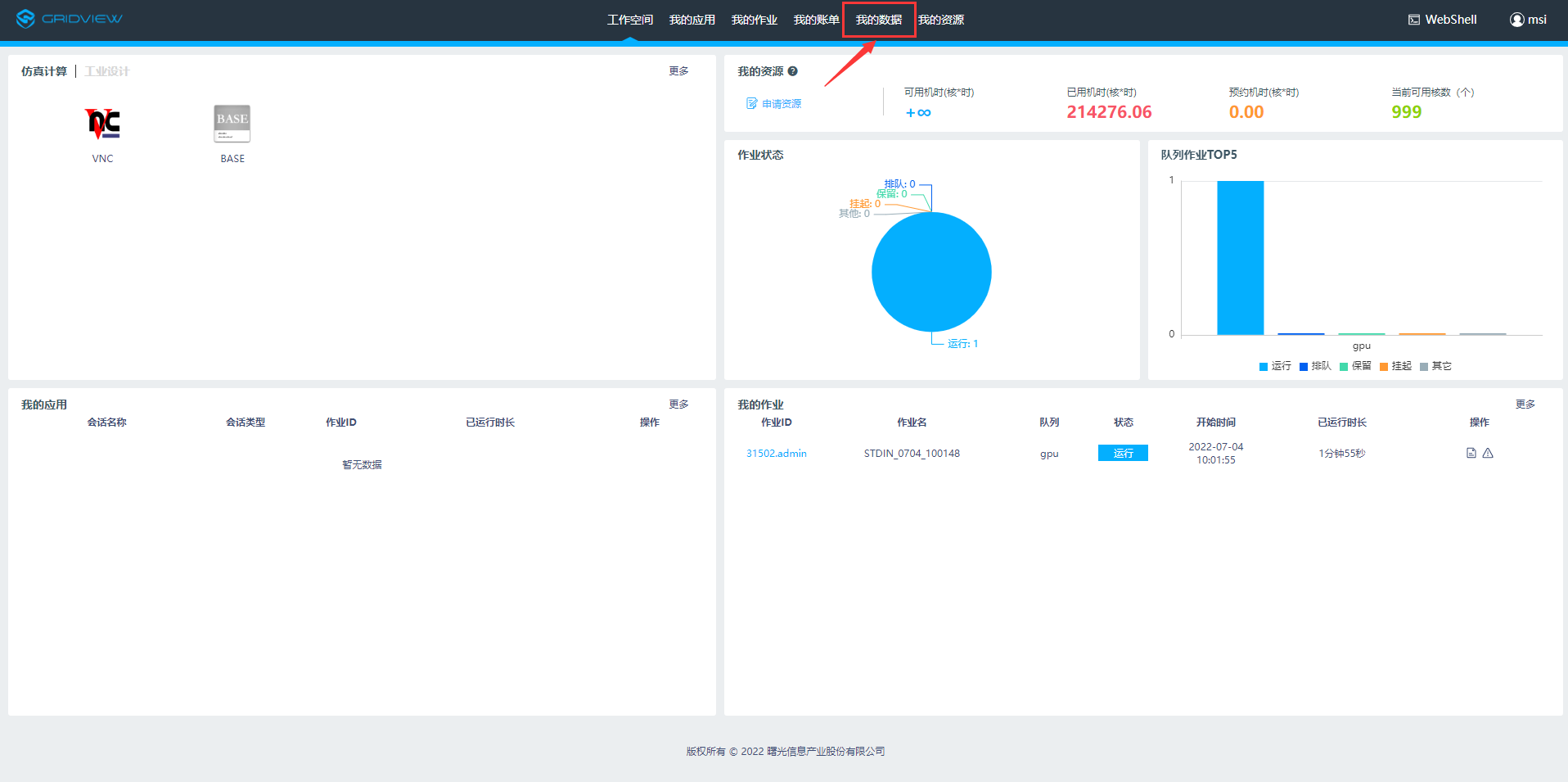


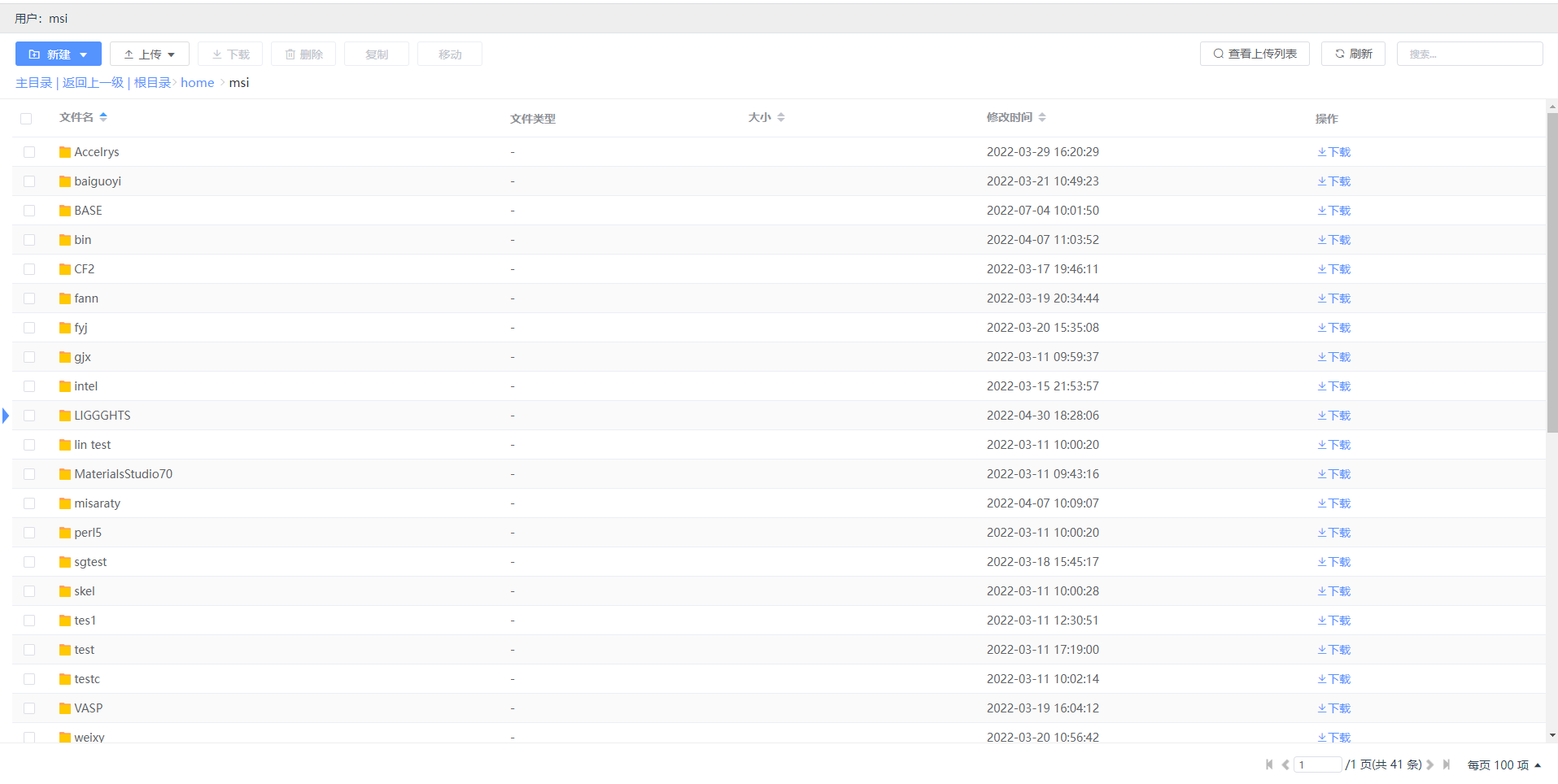


提交申请后，等待系统管理员审批通过，即可使用申请的新的资源量。

2.3.1.4 用户通过网页界面管理用户home目录的数据

用户可以通过web界面的方式上传下载以及删除本地数据。



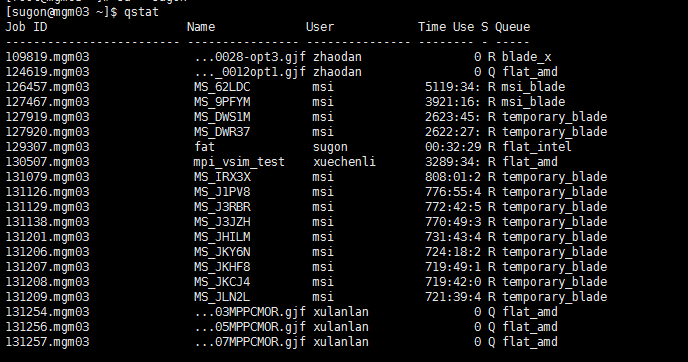


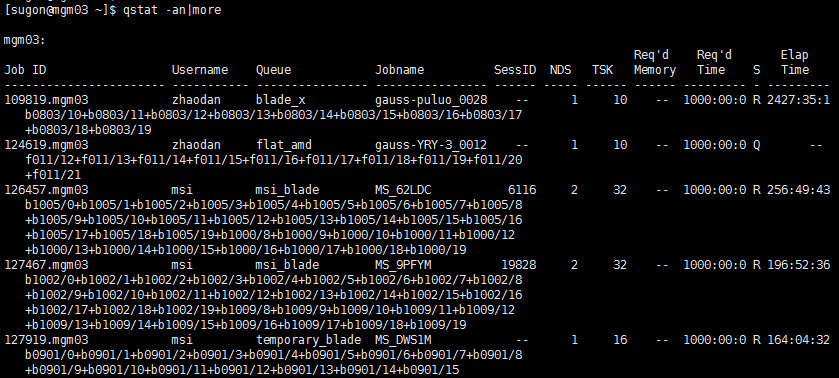
* + 1. 用命令行方式管理作业

2.3.2.1 用户作业状态查询

提交过作业后，用户可以查看作业状态：

qstat





Job Id 作业标识符 PBS自动指定

Username 用户名

Queue 队列名

Jobname 作业名 由作业提交人规定的名称

SessID Session 标识符 仅当作业运行时有值

NDS 使用的节点数

TSK 使用的cpu数或task数

Req’d Memory 作业所需的内存数

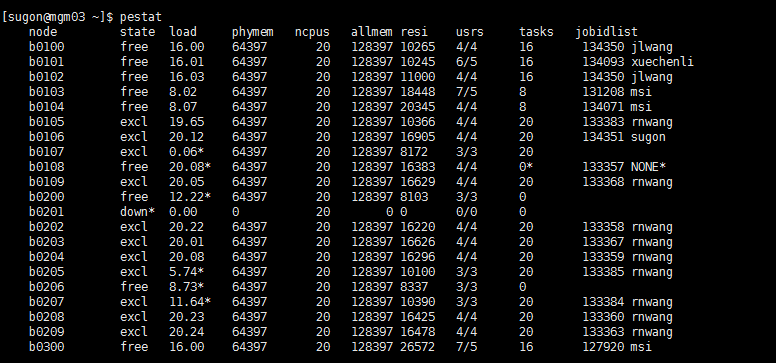
Req’d Time 作业所需的cpu时间或wall time

S 作业状态

Elap Time 作业已经运行的时间

查看节点状态：

pestat



node 节点名

state 节点状态

Load 节点负载

Phymem 物理内 MB

ncpus cpu/核心数

Allmem 分配内存数

Resi 寄存器数

usrs 用户数

Tasks 作业数

Joblist 作业列表

2.3.2.2 用户作业管理

用户可以对列表中队列进行操作。**注意用户可以看到其他用户作业的运行状态，但用户并不能对其他用户的作业进行操作，仅能对于自己的作业进行操作。**操作方式是点选左上角的快捷按钮，可以删除作业，保留作业，挂起作业，恢复作业，释放作业，以及重新运行作业等。如下图所示：

删除作业

qdel -p jobid

交换作业顺序

qorder 111.node1 112.node1

挂起作业：

qhold 111.node1

取消作业挂起

qrls 111.node1

更改作业运行队列：

qmove high 111.node1

* 1. 源码编译

有些用户需要对自行安装源码编译的软件，或编译自编的代码，需要进行源码编译。目前在高性能计算平台上部署的编译器和编译命令如下表所示：

* + 1. c/fortran 编译器汇总选取

|  |  |  |  |
| --- | --- | --- | --- |
| 编译器 | 安装目录 | 版本 | 相关命令 |
| GNU C | /usr/bin | 4.8.5 | gcc \*.c |
| GNU C++ | /usr/bin | 4.8.5 | g++ \*.cpp |
| GNU Fortran | /usr/bin | 4.8.5 | gfortran \*.f |
| Intel C/C++ | source /public/software/profile.d/compiler\_intel-compiler-2017.5.239.sh /public/software/compiler/intel/intel-compiler-2017.5.239/ | 17.0.5 | icc/icpc \*.c |
| GCC C/C++ | source /public/software2/profile.d/compiler\_intel-2020.1.217.sh /public/software2/compiler/intel/2020.1.217/ | 19.1 | icc/icpc \*.c |
| Cmake | source /public/software2/profile.d/compiler\_cmake-3.20.1.sh /public/software2/compiler/cmake/3.20.1/ | 3.20.1 | cmake \* |
| Intel Fortran | source /public/software/profile.d/compiler\_intel-compiler-2017.5.239.sh /public/software/compiler/intel/intel-compiler-2017.5.239/ | 17.0.5 | ifort \*.f |
| Nvidia CUDA Toolkit | /public/software/cuda-11.2/ | 11.2 | nvcc \*.cu |

* + 1. mpi 编译器汇总选取

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MPI实现方式 | 安装目录 | 版本 | MPI编译器 | |
|  |  |  | C编译器 | Fortran编译器 |
| Open mpi | source /public/software/profile.d/mpi\_openmpi-intel-2.1.2.sh | 2.1.2 | mpicc mpicxx | mpif77 mpif90 |
| Open mpi | source /public/software2/profile.d/mpi\_openmpi-intel-4.0.3.sh | 4.0.3 | mpicc mpicxx | mpif77 mpif90 |
| Intel mpi | source /public/software/profile.d/mpi\_intelmpi-2017.4.239.sh | 2017.4.239 | mpicc mpicpc | mpif77 mpif90 |
| Intel mpi | source /public/software2/profile.d/mpi\_intelmpi-2020.1.217.sh | 2020.1.217 | mpicc mpigcc mpicxx mpigxx | mpif77 mpif90 mpifc |

* + 1. 环境变量的选取

**超算中心集群系统数学库环境变量的选取**

|  |  |  |
| --- | --- | --- |
| 数学库 | 版本 | 环境变量选取命令 |
| fftw | float 3.3.7 | source /public/software/profile.d/mathlib\_fftw-intelmpi-3.3.7\_double.sh |
|  | double 3.3.7 | source /public/software/profile.d/mathlib\_fftw-intelmpi-3.3.7\_double.sh |
|  | float 3.3.9 | source /public/software2/profile.d/mathlib\_fftw-intelmpi-3.3.9\_double.sh |
|  | double 3.3.9 | source /public/software2/profile.d/mathlib\_fftw-intelmpi-3.3.9\_double.sh |
| lapack | intel 3.4.2 | source /public/software/profile.d/mathlib\_fftw-lapack-intel-3.4.2.sh |
|  | gnu 3.9.1 | source /public/software2/profile.d/mathlib\_lapack-gnu-3.9.1.sh |
|  | intel 3.9.1 | source /public/software2/profile.d/mathlib\_lapack-intel-3.9.1.sh |
| hdf5 | 1.8.12 | source /public/software/profile.d/mathlib\_hdf5-intel-1.8.12.sh |
|  | 1.12.0 | source /public/software2/profile.d/mathlib\_hdf5-intel-1.12.0.sh |
| gsl | intel 2.6 | source /public/software2/profile.d/mathlib\_gsl-gnu-2.6.sh |
|  | gnu 2.6 | source /public/software2/profile.d/mathlib\_gsl-intel-2.6.sh |
| jasper | gnu 1.900.1 | source /public/software2/profile.d/mathlib\_jasper-gnu-1.900.1.sh |
|  | intel 1.900.1 | source /public/software2/profile.d/mathlib\_jasper-intel-1.900.1.sh |
| netcdf | intel 4.1.3 | source /public/software/profile.d/mathlib\_netcdf-intel-4.1.3.sh |
|  | gnu 4.7.4 | source /public/software2/profile.d/mathlib\_netcdf-gnu-4.7.4.sh |
|  | intel 4.7.4 | source /public/software2/profile.d/mathlib\_netcdf-intel-4.7.4.sh |
| petsc | 3.15.0 | source /public/software2/profile.d/mathlib\_petsc-intelmpi-3.15.0.sh |

附录1 应用软件安装信息

|  |  |  |  |
| --- | --- | --- | --- |
| 软件名 | 版本 | 安装路径 | 备注 |
| VASP | 5.4.1 | /public/software/apps/vasp/intelmpi/5.4.1/ vasp\_std、vasp\_gam、vasp\_ncl |  |
|  | 5.4.4 | /public/software2/apps/vasp/intelmpi/5.4.4/ |  |
| MaterialsStudio | 7 | /home/msi/MaterialsStudio70/ | su - msi进行管理 |
| Gaussian | B01 | /public/software/g09 | source /public/software /profile.d/g09-env.sh 使用 |
| MATLAB | R2018a | /public/software/Matlab/bin/matlab |  |
| phonopy | 1.9.2 | /public/software/phonopy |  |
| lammps | 14May16 | /public/software/apps/lammps/intelmpi/14May16/lmp\_intelmpi |  |
|  | 29Oct20 | /public/software2/apps/lammps/intelmpi/29Oct20/external\_libs/ |  |
| anaconda3 | 2021.05 | /public/software2/apps/anaconda3/2021.05 |  |
| abinit | 8.6.1 | /public/software/apps/abinit/intelmpi/8.6.1/bin |  |

附录2 队列信息

注意！！请勿超过最大nodes、ppn和walltime的限制！

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 队列名称 | 最大可用节点数nodes | 单节点最大可用核数ppn | 作业最小使用核数nodes | 作业最小使用节点数  ppn | 可访问节点数 |
| debug | 62 | 64 | 1 | 1 | 62 |
| gpu | 4 | 40 | 1 | 1 | 4 |
| gauss\_blade | 9 | 20 | 1 | 1 | 9 |
| gauss\_flat | 3 | 64 | 1 | 1 | 3 |
| vasp\_blade | 22 | 20 | 1 | 1 | 22 |
| test\_flat | 3 | 64 | 1 | 1 | 3 |
| other\_flat | 3 | 64 | 1 | 1 | 3 |
| vasp\_flat | 7 | 64 | 1 | 1 | 7 |
| other\_blade | 12 | 20 | 1 | 1 | 12 |
| test\_blade | 6 | 20 | 1 | 1 | 6 |
| new\_c | 5 | 96 | 1 | 1 | 5 |

附录3 Material-Studio客户端配置

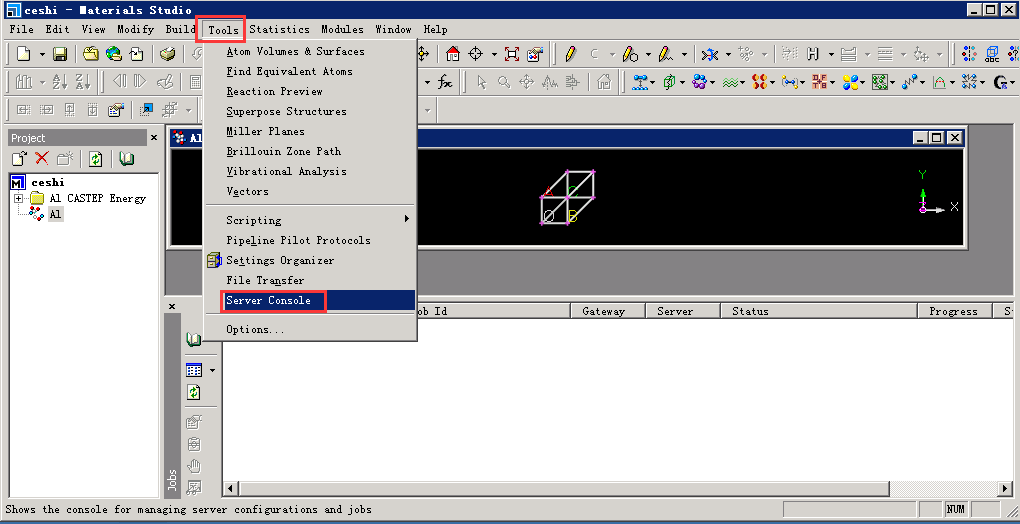
高性能计算平台MaterialsStudio7.0的Gateway采用基于SSL的加密HTTPS传输协议，必须通过MaterialsStudio用户名和密码控制远程MaterialsStudio客户端。

申请开通MaterialsStudio7.0软件使用权限的用户请向河北大学信息技术中心申请，申请通过后由信息技术中心创建MaterialsStudio用户、密码、分配可用队列、最大可用核心。

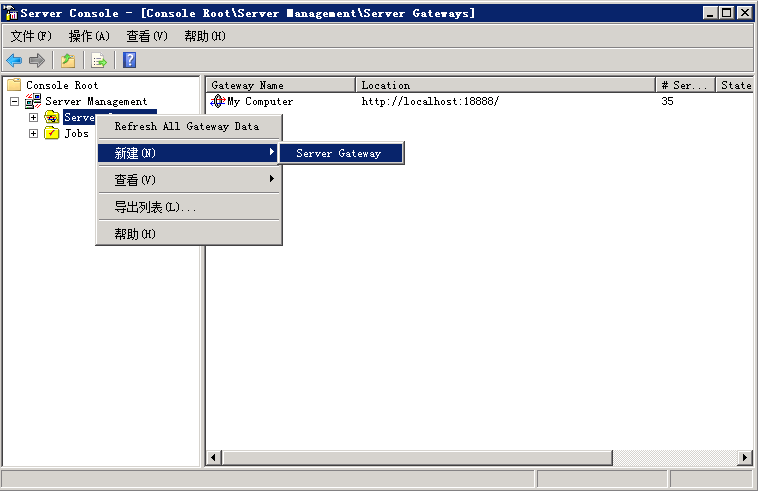
用户申请通过后按照本文提示添加Server Gateway后，可使用Gateway提交作业至高性能计算平台。

3. 1. 添加Server Gateway

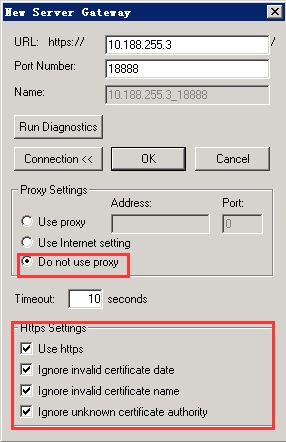
在菜单中选择Tools，然后点击Server Console



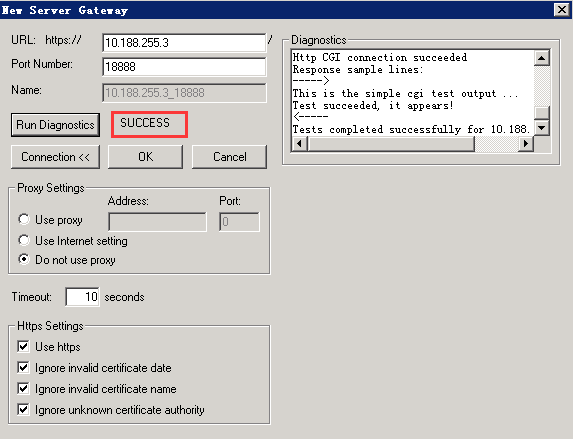
进入Server Console后，选择Server Gateways-> 新建-> Server Gateway



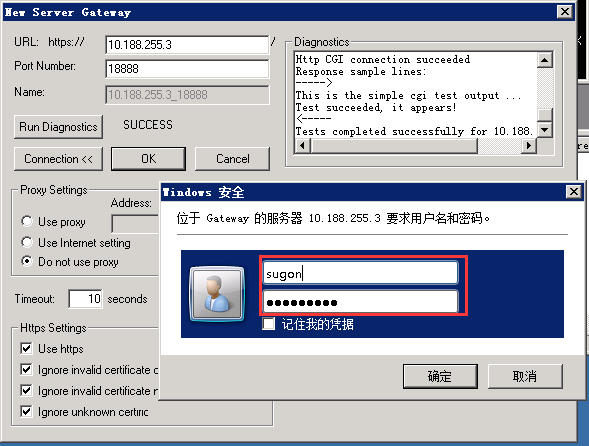
进入Server Gateway对话框后，在URL栏输入IP地址10.188.255.3，Port Number栏中入端口号18888，点击Connection后，选择Do not use proxy，Https Settings全部选择勾选

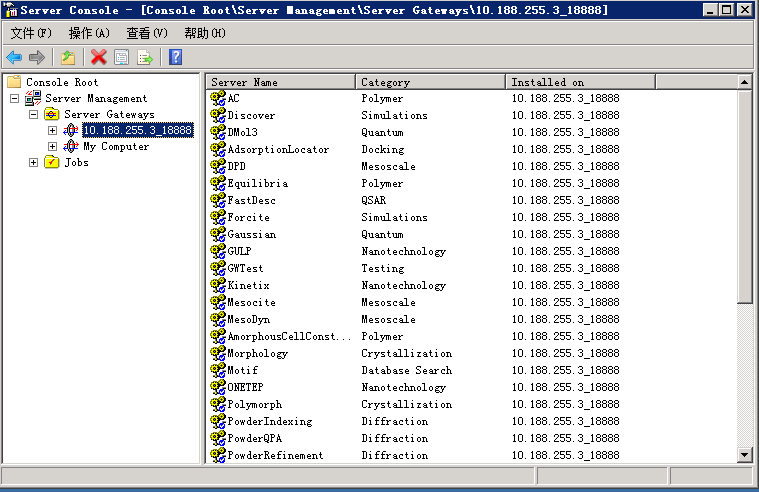


点击Run Diagnostics进行测试，显示SUCCESS则表示测试成功。



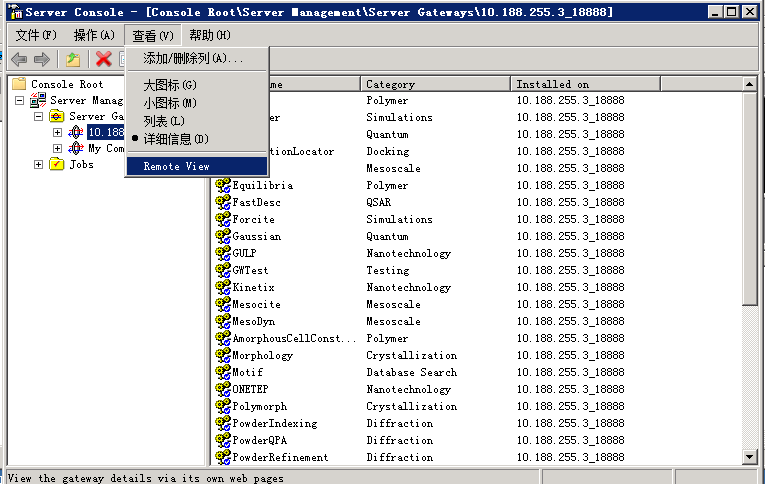
然后点击OK按钮。输入信息技术中心分配的用户名和密码，至此，Server Gateway添加成功。





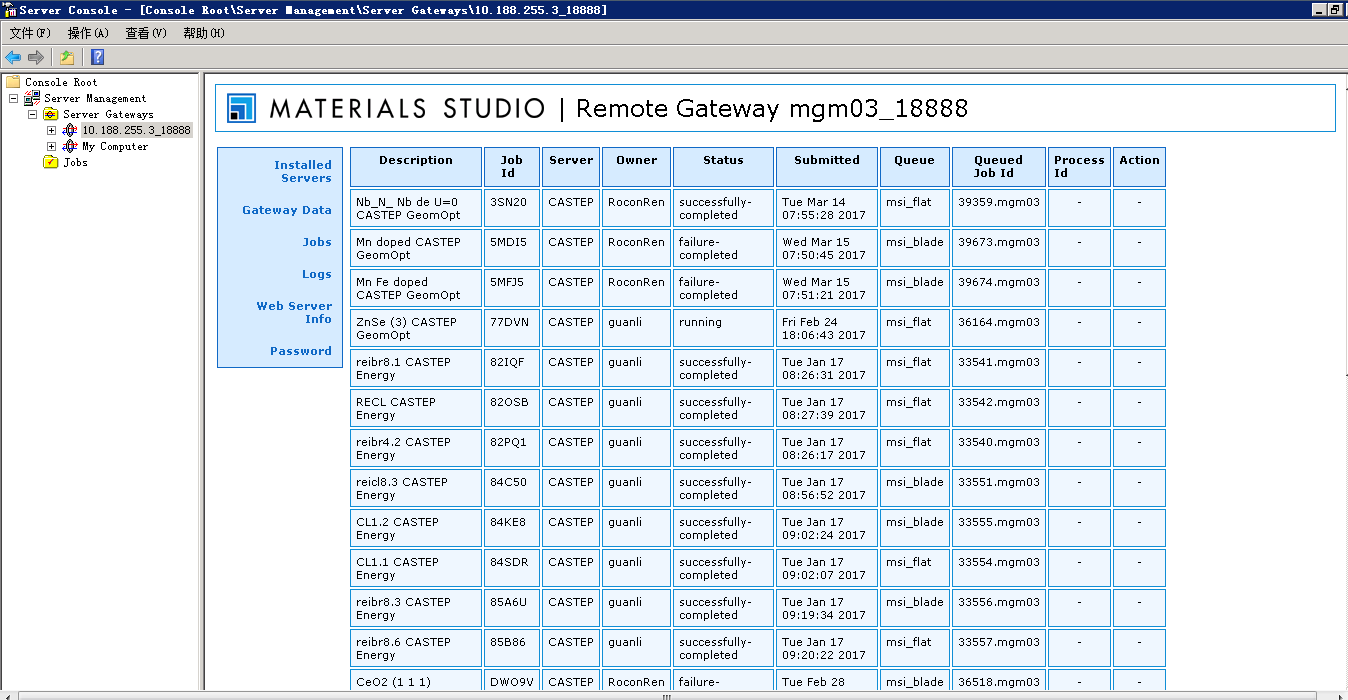
* 1. 查看Gateway的配置信息

在Server Console菜单栏中选择查看->Remote View，则可查看Gateway配置作业，日志等信息。





在jobs中能查看到服务端所有的作业信息，在最右侧的列Action中还可以对作业进行停止、移除等操作。

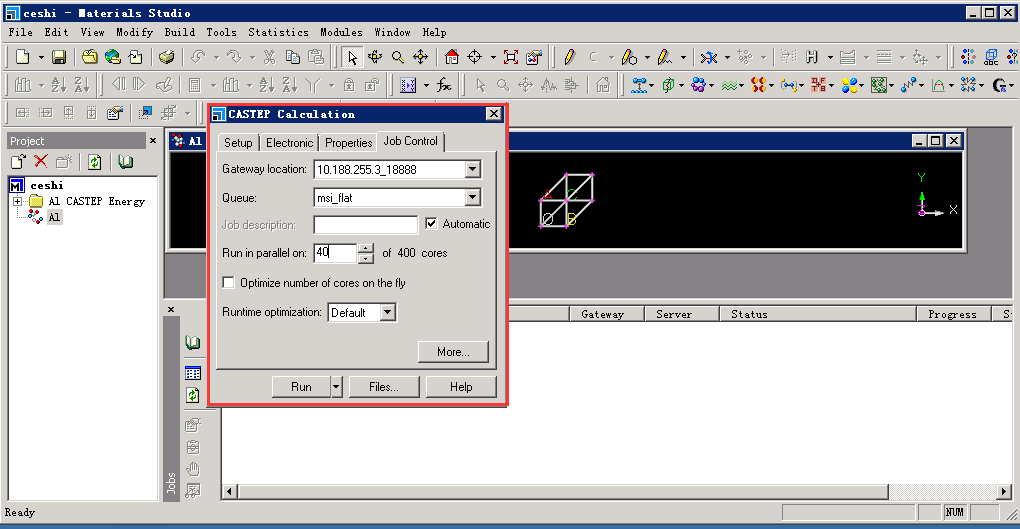


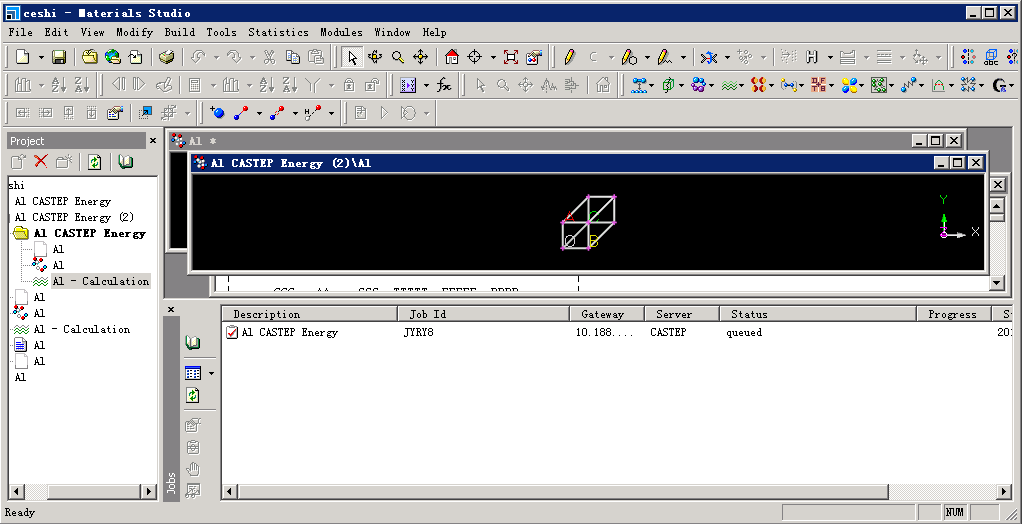
* 1. 使用Gateway向服务端提交作业

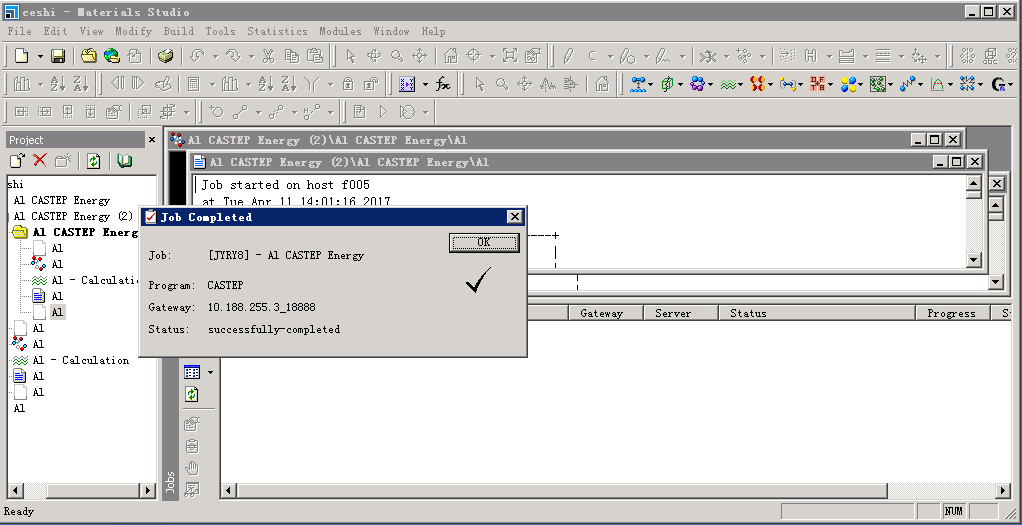
以Castep为例，进入Job Control，在Gateway location下拉菜单中选择开通的Gateway，在Queue下拉菜单中选择队列（由信息技术中心指定），在Run in parallel on中选择核心数。

点击Run按钮即将作业提交至河北大学超算中心集群进行计算。

在MS界面下方的jobs栏中会显示本台计算机提交的作业信息（与Server Console不同，Server Console会显示出在服务端的所有作业的信息）。







附录4 常用软件pbs脚本模板

1. 1. vasp-5.4.1-openmpi版本

#!/bin/bash

#

#PBS -N carbon #请修改作业名

#PBS -q blade\_x #请修改队列

#PBS -l nodes=2:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数

#PBS -l walltime=600:00:00 #指定作业最长运行时间

#PBS -j oe

#PBS -V

source /public/software/profile.d/apps\_vasp-openmpi-5.4.1.sh

# setup hostfile

echo "host: "

# cat $PBS\_JOBID-$PBS\_JOBCOOKIE.hosts

echo "--------Begin-------------------------------------------"

# This job's working directory

echo Working directory is $PBS\_O\_WORKDIR

# go to work dir

cd $PBS\_O\_WORKDIR

echo "Beginning time is: "

echo Time is `date`

echo Directory is `pwd`

NP=`cat $PBS\_NODEFILE|wc -l`

echo "Numbers of Processors: $NP"

echo "---------------------------------------------------"

# running

time mpirun -np $NP -machinefile $PBS\_NODEFILE --mca btl self,sm,openib --bind-to-core $EXEC>> test.out

# clean

echo "--------Finish-------------------------------------------"

echo "Finishing time is: "

echo Time is `date`

echo "--------Finish-------------------------------------------"

* 1. vasp-5.2.12-openmpi版本

#!/bin/bash

#

#PBS -N carbon #请修改作业名

#PBS -q blade\_x #请修改队列

#PBS -l nodes=2:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数

#PBS -l walltime=600:00:00 #指定作业最长运行时间

#PBS -j oe

#PBS -V

source /public/software2/profile.d/apps\_vasp-openmpi-5.4.4.sh

# setup hostfile

echo "host: "

# cat $PBS\_JOBID-$PBS\_JOBCOOKIE.hosts

echo "--------Begin-------------------------------------------"

# This job's working directory

echo Working directory is $PBS\_O\_WORKDIR

# go to work dir

cd $PBS\_O\_WORKDIR

echo "Beginning time is: "

echo Time is `date`

echo Directory is `pwd`

NP=`cat $PBS\_NODEFILE|wc -l`

echo "Numbers of Processors: $NP"

echo "---------------------------------------------------"

# running

time mpirun -np $NP -machinefile $PBS\_NODEFILE --mca btl self,sm,openib --bind-to-core $EXEC>> test.out

# clean

echo "--------Finish-------------------------------------------"

echo "Finishing time is: "

echo Time is `date`

echo "--------Finish-------------------------------------------"

* 1. MaterialsStudio7.0

#PBS -N Fe2O3\_120 #请修改作业名

#PBS -q blade\_x #请修改队列

#PBS -j oe

#PBS -l nodes=2:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数

#PBS -l walltime=100:00:00 #指定作业最长运行时间

cd $PBS\_O\_WORKDIR

start\_time=$(date +%s)

echo "start time: $(date)"

echo "Running nodes:"

cat $PBS\_NODEFILE

echo "Total cores: $(cat $PBS\_NODEFILE|wc -l)"

NP=$(cat $PBS\_NODEFILE|wc -l)

model=$(ls \*.param|head -n 1)

model=${model%%.\*}

export MS\_INSTALL\_ROOT=/home/msi/Accelrys/MaterialsStudio7.0

eval `${MS\_INSTALL\_ROOT}/share/license/data/lic\_setup.sh ${MS\_INSTALL\_ROOT} -s sh`

eval `${MS\_INSTALL\_ROOT}/share/bin/ms\_setup.sh ${MS\_INSTALL\_ROOT} -s sh`

export PSPOT\_DIR=${MS\_INSTALL\_ROOT}/share/Resources/Quantum/Castep/Potentials

export DMOL3\_DATA=${MS\_INSTALL\_ROOT}/share/Resources/Quantum/DMol3

export DMOL\_TMP=/tmp

mpirun -np $NP -machinefile $PBS\_NODEFILE /home/msi/Accelrys/MaterialsStudio7.0/bin/castepexe.exe $model

echo "end time: $(date)"

end\_time=$(date +%s)

echo "Job Running Time:$((end\_time - $start\_time))"

* 1. Gaussian09

#PBS -N gaussian #请修改作业名

#PBS -q blade\_x #请修改队列

#PBS -l nodes=1:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数,不推荐做跨节点运算

#PBS -j oe

#PBS -l walltime=1000:00:00 #指定作业最长运行时间

cd $PBS\_O\_WORKDIR

JOB=h2o.com #指定输入文件名

JOBNAME=`basename "$JOB" .com`

STARTTIME=`date` #获取作业开始时间

STIME=`date +%s`

source /public/software/profile.d/g09-env.sh #读取环境变量

/public/software/g09/g09 < $JOB > "$JOBNAME.log" #开始计算

echo >>"$JOBNAME.log"

echo g09 job started: $STARTTIME >>"$JOBNAME.log" #输出作业的时间信息

echo g09 job finished: `date` >>"$JOBNAME.log"

let ETIME=`date +%s`-$STIME

echo Elapsed time: $ETIME >>"$JOBNAME.log"

* 1. Phonopy

phonopy现在一般用作与vasp链接使用，或者和MS链接使用。在pbs中的提交方式如下：具体phonopy\* 所需要执行的指令和参数根据您具体所需要运行的作业而定。

目前的指令有：phonopy、phonopy-FHI-aims、phonopy-qha

#PBS -N phonopy #请修改作业名

#PBS -q blade\_x #请修改队列

#PBS -l nodes=1:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数,不推荐做跨节点运算

#PBS -j oe

#PBS -l walltime=1000:00:00 #指定作业最长运行时间

cd $PBS\_O\_WORKDIR

JOB=h2o.com #指定输入文件名

JOBNAME=`basename "$JOB" .com`

STARTTIME=`date` #获取作业开始时间

STIME=`date +%s`

source /public/software/profile.d/apps\_phonopy-1.9.2.sh #读取环境变量

mpirun -np $NP lmp\_mpi -in phonopy\* -\* | tee -a $LOG\_FILE #开始计算

echo >>"$JOBNAME.log"

echo g09 job started: $STARTTIME >>"$JOBNAME.log" #输出作业的时间信息

echo g09 job finished: `date` >>"$JOBNAME.log"

let ETIME=`date +%s`-$STIME

echo Elapsed time: $ETIME >>"$JOBNAME.log"

* 1. Matlab

#PBS -N matlab #请修改作业名

#PBS -q debug #请修改队列

#PBS -l nodes=1:ppn=20 #nodes为请求的节点数，ppn为每节点使用的核数,不推荐做跨节点运算

#PBS -j oe

#PBS -l walltime=1000:00:00 #指定作业最长运行时间

cd $PBS\_O\_WORKDIR

STARTTIME=`date` #获取作业开始时间

STIME=`date +%s`

source /public/software/profile.d/apps\_matlab-R2018a.sh #读取环境变量

mpirun -np $NP lmp\_mpi -in matlab -\* | tee -a $LOG\_FILE #开始计算

echo >>"$JOBNAME.log"

echo g09 job started: $STARTTIME >>"$JOBNAME.log" #输出作业的时间信息

echo g09 job finished: `date` >>"$JOBNAME.log"

let ETIME=`date +%s`-$STIME

echo Elapsed time: $ETIME >>"$JOBNAME.log"