

# 从头开始用 VASP 做结构优化

张旻烨 王越超 许熙 栾东

登入 **Linux**

编辑文件

**VASP** 输入文件解读

执行 **VASP** 计算

解读结构优化过程

## 登入 Linux

# 计算机通信协议

- ▶ 安全 Shell 协议 (SSH)
- ▶ 远程桌面类型 (VNC, RDP)
- ▶ 文件传输协议 (FTP)

# 基于 SSH 协议的 SSH 客户端

## 基本需求

1. 通过命令行的方式操作远端的电脑 (投任务, 编辑文本, 编译程序, ...)
2. 上传下载文件
3. 支持 Linux 上的窗口化的程序 (比如 [xmgrace](#))

# 微软 Windows 下的 SSH 客户端

## 1. 自带的 ssh 客户端工具

优点: Windows10 原生程序, 不需要第三方的程序

缺点: 文件互传不便, 默认不支持 Linux 上窗口化程序

## 2. 第三方 ssh 客户端, 例如MobaXterm

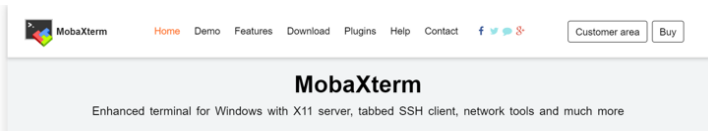


Figure 1: MobaXterm

优点: 免费软件, 拖拽式文件上传下载, 支持 X Windows 窗口软件

缺点: 并不是总能跟上 Windows 的更新频率

# 自带 SSH 客户端 I

← 设置

## 管理可选功能

可选功能

[查看可选功能历史记录](#)



添加功能



Internet Explorer 11

1.74 MB



Microsoft 快速助手

1.52 MB



OpenSSH 客户端

5.70 MB



Windows Media Player

61.1 MB



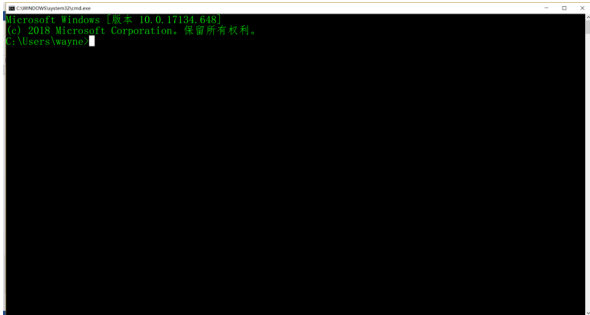
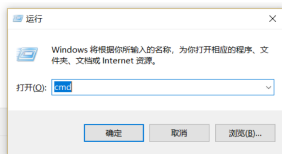
XPS Viewer

16.1 MB

Figure 2: 安装 SSH 客户端

# 自带 SSH 客户端 II

打开 Windows 的命令提示符窗口: Win+R → cmd



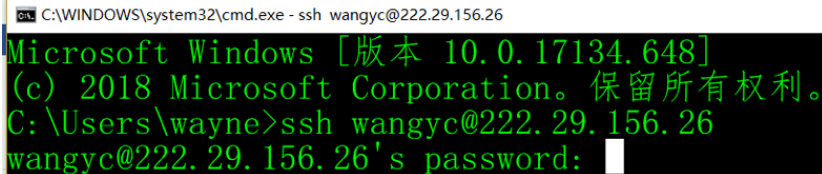


# 自带 SSH 客户端 III

## 1. 输入 ssh 登录的命令

@ 前面是用户名, @ 后面是远端服务器的 IP

## 2. 输入密码



A screenshot of a Windows command prompt window. The title bar shows the path 'C:\WINDOWS\system32\cmd.exe' and the command 'ssh wangyc@222.29.156.26'. The command prompt displays the following text in green on a black background: 'Microsoft Windows [版本 10.0.17134.648]', '(c) 2018 Microsoft Corporation。保留所有权利。', 'C:\Users\wayne>ssh wangyc@222.29.156.26', and 'wangyc@222.29.156.26's password:'. A white cursor is visible at the end of the password prompt.

Figure 3: 登录命令

# 自带 SSH 客户端 IV

登录成功! 可以在命令行下控制远端设备了

wangyc@b214: ~

```
C:\Users\wayne>ssh wangyc@222.29.156.26
```

```
wangyc@222.29.156.26's password:
```

```
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-47-generic x86_64)
```

- \* Documentation: <https://help.ubuntu.com>
- \* Management: <https://landscape.canonical.com>
- \* Support: <https://ubuntu.com/advantage>

- \* Ubuntu's Kubernetes 1.14 distributions can bypass Docker and use containerd directly, see <https://bit.ly/ubuntu-containerd> or try it now with

```
    snap install microk8s --classic
```

- \* Canonical Livepatch is available for installation.
  - Reduce system reboots and improve kernel security. Activate at: <https://ubuntu.com/livepatch>

```
82 packages can be updated.
```

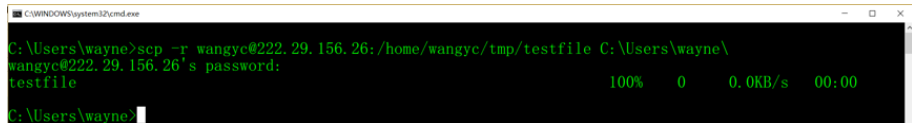
```
0 updates are security updates.
```

```
Last login: Mon Apr  8 18:50:38 2019 from 222.29.156.16
```

```
wangyc@b214: ~$
```

# 自带 SSH 客户端 V

传输文件: 使用 `scp` 命令



```
C:\WINDOWS\system32\cmd.exe
C:\Users\wayne>scp -r wangyc@222.29.156.26:/home/wangyc/tmp/testfile C:\Users\wayne\
wangyc@222.29.156.26's password:
testfile
100% 0 0.0KB/s 00:00
C:\Users\wayne>
```

含义:

将用户 wangyc 在

222.29.156.26 机器上的 `/home/wangyc/tmp/testfile` 文件

拷贝到 `C:\Users\wayne\` 目录下

# 自带 SSH 客户端 VI

免密码登录: 使用 `ssh-keygen` 命令

一路回车下去, 创建公钥和私钥

在远端服务器上已进行相同的操作, 产生 `~/.ssh/` 目录

```
C:\WINDOWS\system32\cmd.exe

C:\Users\wayne>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\wayne\.ssh\id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\wayne\.ssh\id_rsa.
Your public key has been saved in C:\Users\wayne\.ssh\id_rsa.pub.
The key fingerprint is:
SHA256:REcZFsio/HUj4fjMLm6qC9I7jpuApUR45o4X9MPFMHM wayne@my-laptop
The key's randomart image is:
+---[RSA 2048]-----+
|
| o+oE..
| . . oB =
| oo+ o * .
| o* + * +
| . = 0 o S
| o* o =
| B + .
| ++. + .
| B=*o.
+---[SHA256]-----+

C:\Users\wayne>
```

# 自带 SSH 客户端 VII

将 .ssh 目录下的 id\_rsa.pub 上传到远端服务器

C:\WINDOWS\system32\cmd.exe

```
C:\Users\wayne>cd .ssh/
```

```
C:\Users\wayne\.ssh>dir
```

驱动器 C 中的卷是 OS  
卷的序列号是 826C-9374

C:\Users\wayne\.ssh 的目录

```
2019/04/08  19:28    <DIR>          .
2019/04/08  19:28    <DIR>          ..
2019/04/08  19:23             1,679 id_rsa
2019/04/08  19:23             398 id_rsa.pub
2018/12/21  19:27             2,121 known_hosts
                3 个文件             4,198 字节
                2 个目录    6,832,545,792 可用字节
```

```
C:\Users\wayne\.ssh>scp id_rsa.pub wangyc@222.29.156.26:~/
```

# 自带 SSH 客户端 VIII

```
wangyc@b214: ~/.ssh$ cat ~/.ssh/id_rsa.pub >> authorized_keys
```

即可无密码访问远端服务器

# MobaXterm I

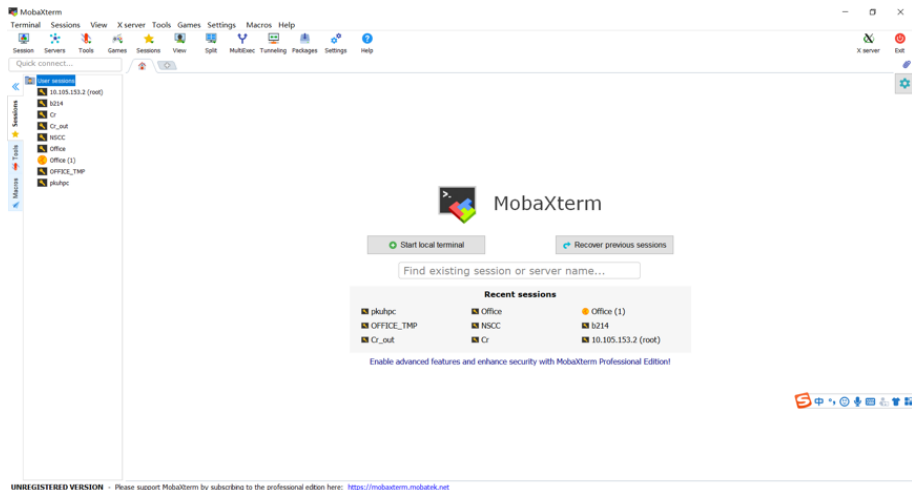


Figure 4: MobaXterm 界面

# MobaXterm II

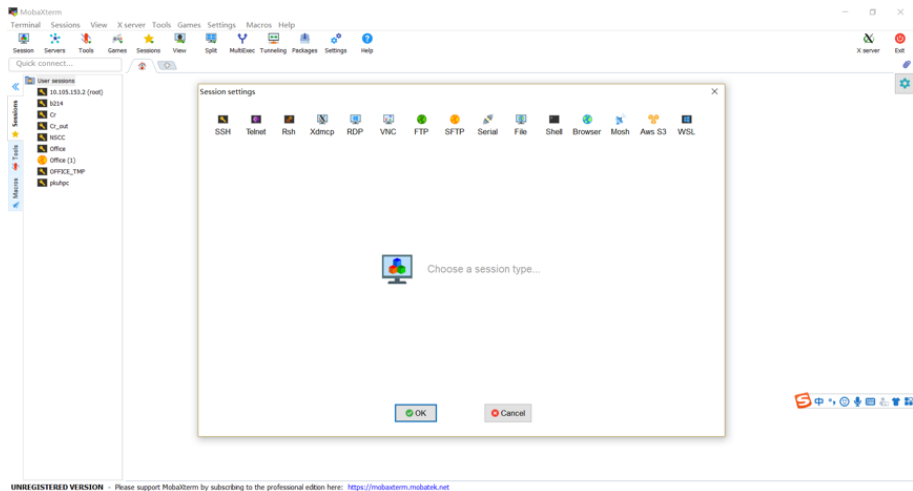
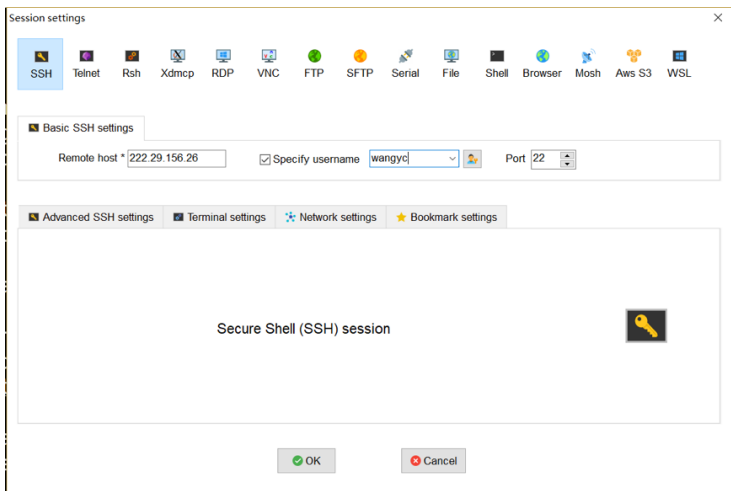


Figure 5: sessions → new session 对话框

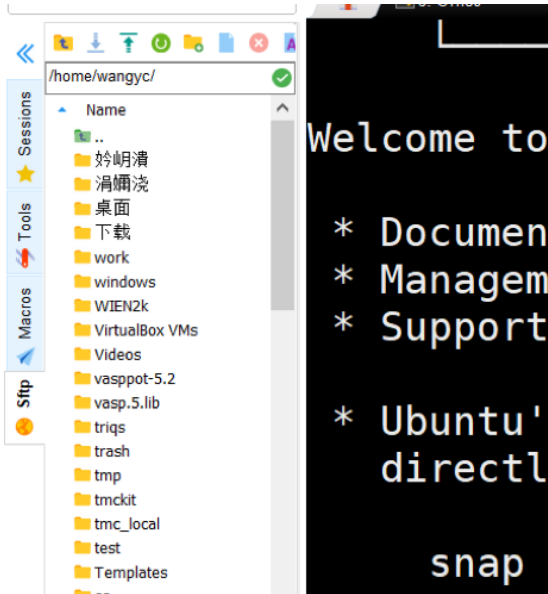


# MobaXterm III

选择中间对话框左上角的 SSH 标签, 填好远端的 IP 和用户名, 点击 OK, 第一次登陆会询问是否记住密码, 选择记住之后登陆就不用输入密码了



# MobaXterm IV



登陆之后左侧出现文件列表  
可以通过拖拽的方式进行文件  
的上传和下载

## 编辑文件

# Linux 下常用文本编辑器: Vim

- ▶ 语法高亮
- ▶ 简单的自动补全
- ▶ 列编辑和多文件编辑
- ▶ 通过键盘输入命令实现, 同时支持鼠标操作

# 学习 Vim

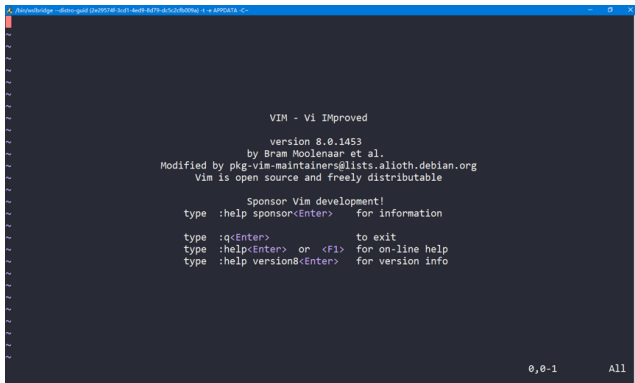
- ▶ 善用搜索引擎 (百度, Google, ...)

想知道 vim 如何查找替换  $\Rightarrow$  搜 “vim 查找替换”

- ▶ Linux 下各种命令与程序的用法: CSDN 博客, 脚本之家

## 启动,保存与退出 I

命令行输入 `vim`, 回车, 会出现 `vim` 的一个界面

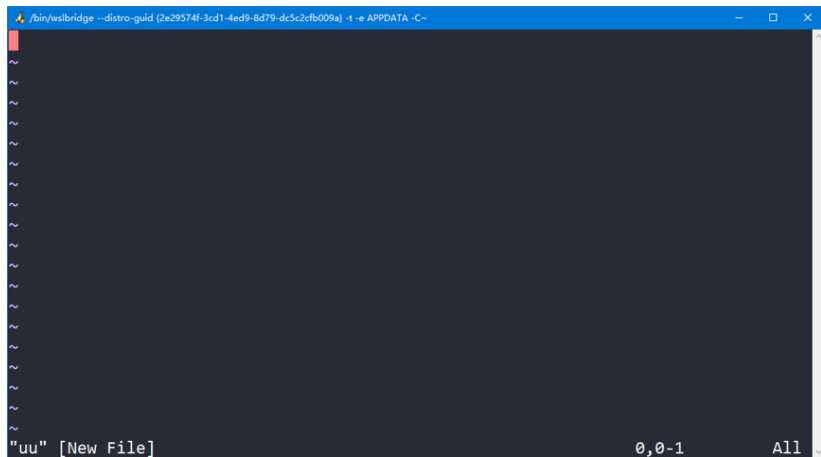


### Figure 6: Vim 界面

如果后面加上文件名就会打开相应的文件,若该文件不存在则创建一个新文件

# 启动, 保存与退出 II

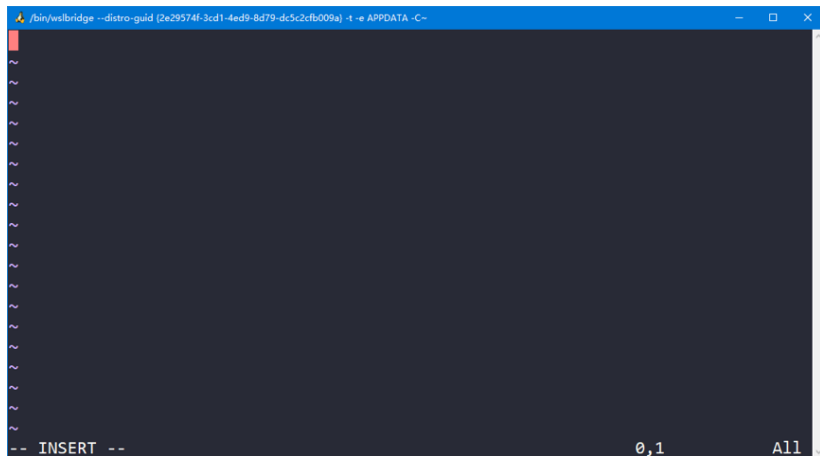
此时你在键盘上的任何输入一般都是无效的



The screenshot shows a terminal window with a blue title bar containing the text `/bin/wslbridge --distro-guid [2e29574f-3cd1-4ed9-8d79-dc5c2cfb009a] -t -e APPDATA -C-`. The main area is a dark-themed file editor. On the left, a vertical list of tilde characters (~) indicates file structure. The bottom status bar shows `"uu" [New File]` on the left, `0,0-1` in the center, and `All` on the right.

# 启动, 保存与退出 III

需要按一下 `i` 键变成可输入状态, 然后向里面输入内容



The screenshot shows a terminal window with a blue title bar. The title bar text is `/bin/wslbridge --distro-guid [2e29574f-3cd1-4ed9-8d79-dc5c2cfb009a] -t -e APPDATA -C-`. The terminal content is a dark blue background with a vertical column of tilde characters (`~`) on the left side. At the bottom left, it says `-- INSERT --`. At the bottom right, it shows `0,1` and `All`.



# 启动, 保存与退出 IV

- ▶ vim 的很多操作是通过“命令”进行的
- ▶ 在--INSERT--状态下, 任何键盘输入都成为了输入的内容



# 启动, 保存与退出 VI

输入命令

- ▶ :wq(write and quit), 回车  $\Rightarrow$  保存并退出
- ▶ :q!, 回车  $\Rightarrow$  直接退出, 不保存

注意: 两个命令都以一个冒号开头

# 复制, 粘贴和删除文本

**Table 1:** Vim 常用命令

命令	作用
yy	复制当前一整行的内容
p	把复制的内容粘贴到光标所在行的下一行
dd	删除光标所在的一整行内
u	撤销上次输入
.	恢复上次输入

- ▶ Vim 不支持鼠标光标选中后 Delete
- ▶ 注意上面这些都是按了 Esc 之后执行, 不需要加冒号, 在--INSERT--状态无法执行

# 文本定位, 查找和替换 I

## 跳转

- ▶ `Shift + G` 跳转到文件末尾
- ▶ 按两次 `[` 可以跳转到文件开头
- ▶ `:n` 跳转到某一行使用命令, `n` 是一个行号
- ▶ 使用命令 `:set nu` 显示行号

# 文本定位, 查找和替换 II

## 查找文本

- ▶ 以 FORCE 为例, 在非插入状态下输入/FORCE, 光标即可跳转到 FORCE 所在的位置
- ▶ 按一下 n 键 (next) 跳转到下一个 FORCE 文本所在的位置
- ▶ 要想从文件末尾查找, 则先按下 Shift + G 跳转到文件末尾, 然后按 Shift + N, 即可跳转到从末尾数的第一个 FORCE 所在位置

# 文本定位, 查找和替换 III

替换: 与 sed 命令类似

- ▶ 在非插入状态下输入命令:`%s/FORCE/force/g`, 把所有 FORCE 替换成 force
- ▶ 如果没有前面的百分号, 则每次只替换一个

# 列编辑 I

非插入状态下按下 **Ctrl + V** 即进入列选择状态

按住方向键上下左右即可按列选择文本

按一下 d 键则删除按列选中的文本

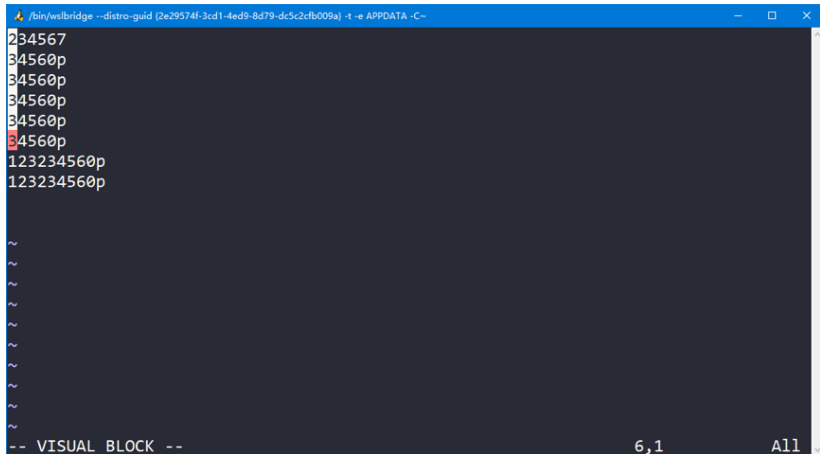
[illegible][illegible]



## 列编辑 II

另一个常用的操作是给很多行加注释

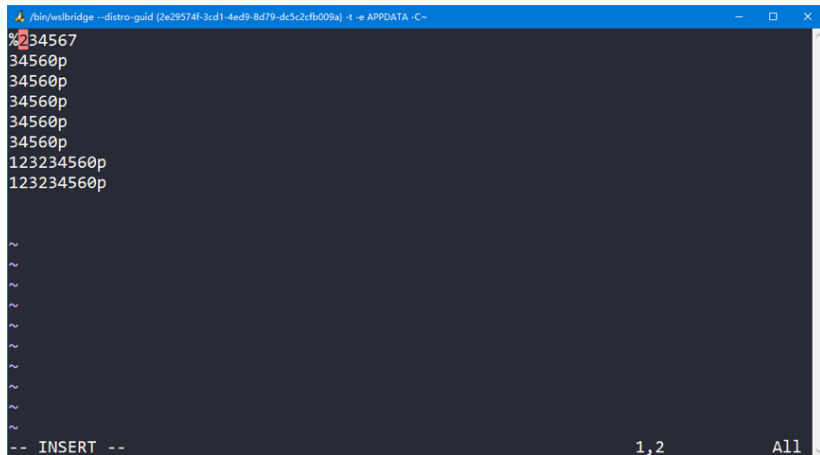
按 `Esc` 键进入非插入状态, 按 `Ctrl + V` 进入列选择状态, 然后选择要插入的列范围



```
/bin/wslbridge --distro-guid [2e29574f-3cd1-4ed9-8d79-dc5c2cfb009a] -t -e APPDATA -C~
234567
34560p
34560p
34560p
34560p
34560p
34560p
123234560p
123234560p
~
~
~
~
~
~
~
~
~
~
-- VISUAL BLOCK --                               6,1                               All
```

# 列编辑 III

按下 Shift + i 键, 输入文本



```
/bin/wslbridge --distro-guid [2e29574f-3cd1-4ed9-8d79-dc5c2cfb009a] -t -e APPDATA -C-
%234567
34560p
34560p
34560p
34560p
34560p
123234560p
123234560p
~
~
~
~
~
~
~
~
~
~
-- INSERT --                               1,2                               All
```

再连按两次 Esc, 前面选择过的行前都出现了相同的内容

# VASP 输入文件解读

# POSCAR: 晶体结构文件

# 获取晶体结构 I

## Inorganic Crystal Structure Database: 实验结构

**ICSD** Welcome to ICSD Web. IP authenticated (222.29.116.46). Peking Univ. FIZ Karlsruhe | Contact Close session

**Login**

LogInID:

Password:

☐ Lost password? ☐ Personalize account

**Content Selection**

☒ Experimental Structures only

☐ Theoretical Structures only

☐ All Structures

**Navigation**

Q Basic search & retrieve

**Advanced search & retrieve**

Q Bibliography

Q Cell

Q Chemistry

Q Symmetry

Q Crystal Chemistry

Q Structure Type

Q Experimental Information

Q DB Info

**Query Management**

☒ Manage Queries

☒ List Combined Queries

☒ Create Combined Query

**Basic Search & Retrieve**

**Bibliography**

Authors  Year of Publication

Title of Journal

Title of Article

**Chemistry**

Composition   Number of Elements

**Cell**

Cell Parameters

Cell Volume  Tolerance +/-  %

**Symmetry**

Space Group Symbol  Space Group Number

Crystal System  Centering

**Exp. Info. & Ref. Data**

New Data Only ☐

PDF Number  Temperature  K

ICSD Collection Code  Pressure  MPa

**Search Action**

**Search Summary**

Basic Search: -

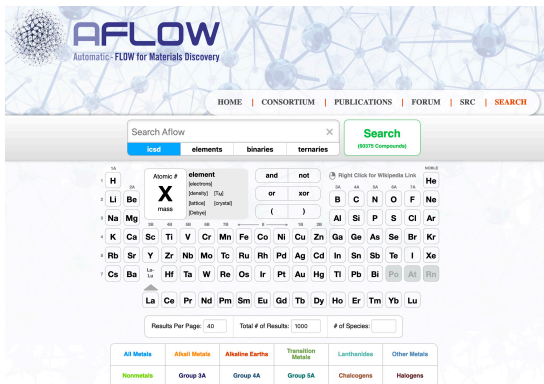
**Query History**

Number of queries: 0

Figure 7: ICSD 搜索页面

# 获取晶体结构 II

**AFLOW**: Duke 材料基因组学数据库



AFLOW  
Automatic- FLOW for Materials Discovery

HOME | CONSORTIUM | PUBLICATIONS | FORUM | SRC | SEARCH

Search Aflow

icad elements binaries ternaries

Search  
(88375 Compounds)

Atomic # element  
mass

and not  
or xor  
( )

Right Click for Wikipedia Link

Results Per Page: 40 Total # of Results: 1000 # of Species:

All Metals	Alkali Metals	Alkaline Earths	Transition Metals	Lanthanides	Other Metals
Nonmetals	Group 3A	Group 4A	Group 5A	Chalcogens	Halogens

Figure 8: AFLOW 搜索页面

# 获取晶体结构 III

Materials Project: 基于 pymatgen 的材料基因组学数据库

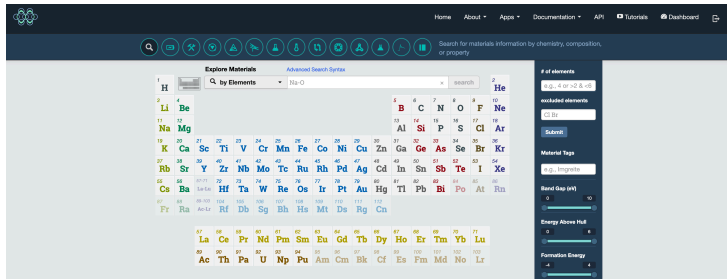


Figure 9: MaterialsProject 搜索页面

## 执行 VASP 计算



# 创建运算目录

## ► 教学一号

```
$ ls
POSCAR POTCAR INCAR KPOINTS sc_run_vasp.sh
$ mkdir session-1/
$ cp POSCAR POTCAR INCAR KPOINTS session-1/
$ cp sc_run_vasp.sh session-1/
$ cd session-1/
```

## ► TMC PC

```
$ ls
Documents Downloads tests ...
$ mkdir -p tests/YOUR_NAME
$ cp POSCAR POTCAR INCAR KPOINTS tests/YOUR_NAME
$ cd tests/YOUR_NAME
```

# 运行 VASP I

确认 vasp 可执行程序的位置: which

## ► 教学一号

```
$ which vasp_std  
$ module load vasp/5.4.4-intel18.0  
$ which vasp_std  
/nfs-share/software/vasp/intel18.0/bin/vasp_std
```

## ► TMC PC

```
$ which vasp_std  
/home/dft003/software/vasp.5.4.4/bin/vasp_std
```

# 运行 VASP II

为什么第一次 `which` 的结果不同?

在教学一号上

```
$ echo $PATH  
/nfs-share/software/vasp/intel18.0/bin/:  
/nfs-share/software/module/bin:/usr/local/bin:/usr/bin:  
...
```

# 运行 VASP III

```
$ cat sc_run_vasp.sh
#!/usr/bin/env bash                                # 解释器
#SBATCH -A 150xxxxxxx                               # 学号
#SBATCH --nodes=1                                    # 使用一个节点
#SBATCH -c 2                                          # 每个任务用 2 个核心
#SBATCH --partition=compute                          # 指定计算分区
#SBATCH -J test                                       # 任务名
#SBATCH -o stdout

module load intel/2018.0                            # 载入 Intel 编译器环境变量
module load vasp/5.4.4-intel18.0                    # 载入 VASP 环境变量 (PATH)

mpirun -np 2 vasp_std                                # 运行
```

# 运行 VASP IV

## ► 教学一号

```
$ sbatch sc_run_vasp.sh  
Submitted batch job xxxx  
$ watch -n 1 cat stdout
```

## ► TMC PC

```
$ mpirun -np 2 vasp_std > out &  
$ watch -n 1 cat out
```

## 解读结构优化过程