# Hands-on introduction of OpenMX
# - Practice -

Institute for Solid State Physics
Mitsuaki Kawamura

Schedule:
10:40～11:10 Lecture
11:10～12:00 Practice

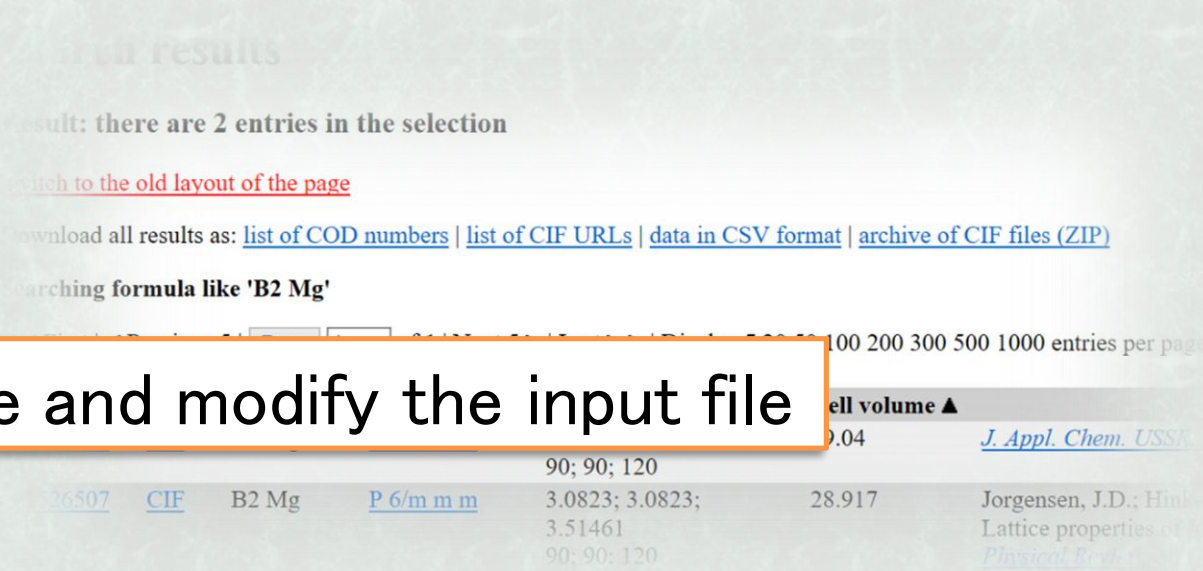
2020/2/14
**The 1st OpenMX hands-on workshop 2020**

# Outline

- Introduction
- Computer in this tutorial
  - Configuration
  - Login
- Input file generation
- Job submit
- Check results
- Post processing and visualization
  - Band structure
  - Kohn–Sham orbitals
  - DOS, PDOS, Fermi surfaces
- Free excise

# Introduction : Purpose

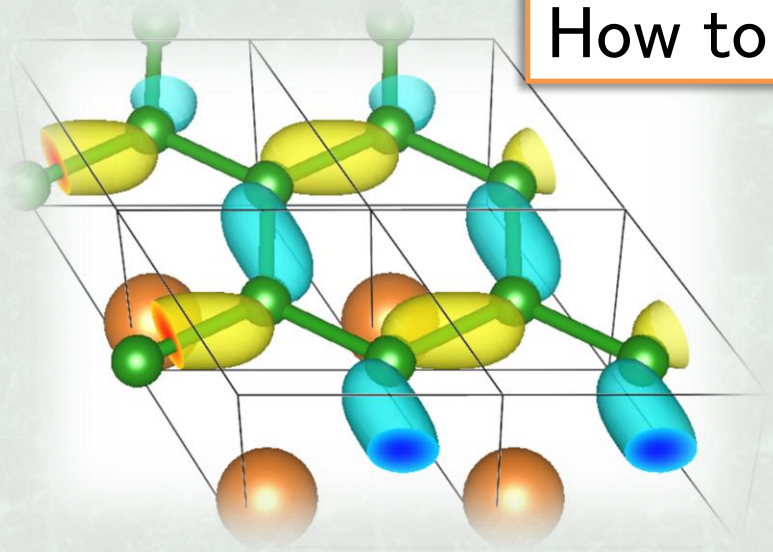
We will overview

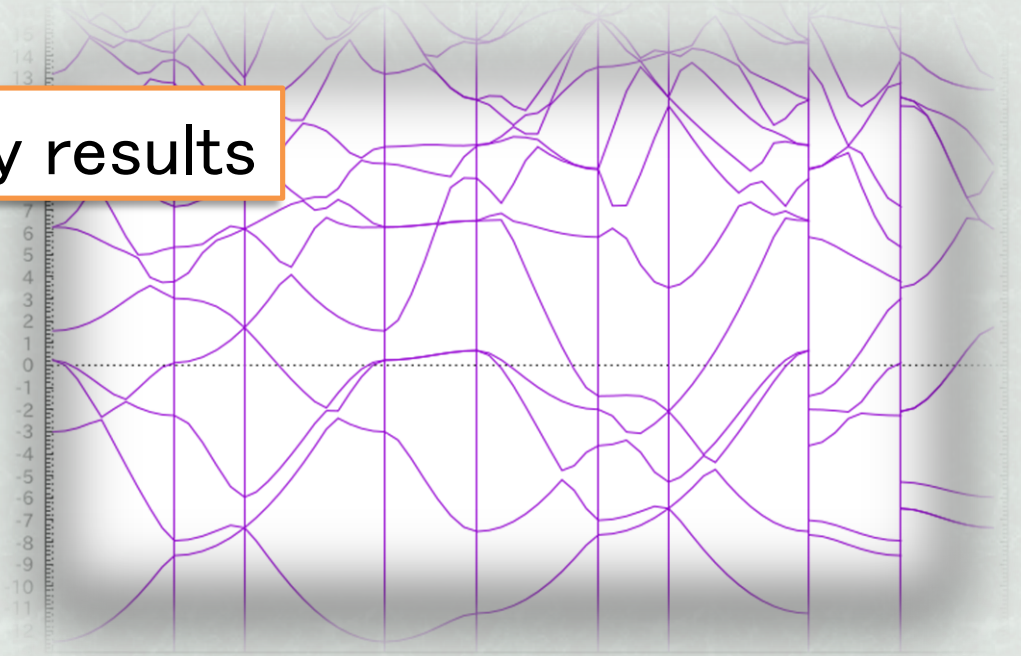



How to generate and modify the input file

and

How to display results





Intro

# Flow of this practice

Log-in workstation

**Input file**

Get CIF file from database → Convert cif to input file → View structure

**SCF**

Submit job for SCF → Check the result

**Band structure**

Modify input for band → Submit job → Plot band structure

**Kohn-Sham orbital**

Modify input for displaying orbital → Submit job → Plot Kohn-Sham orbital

**DOS and Fermi surface**

Modify input for DOS → Submit job → Interactive queue for DOS

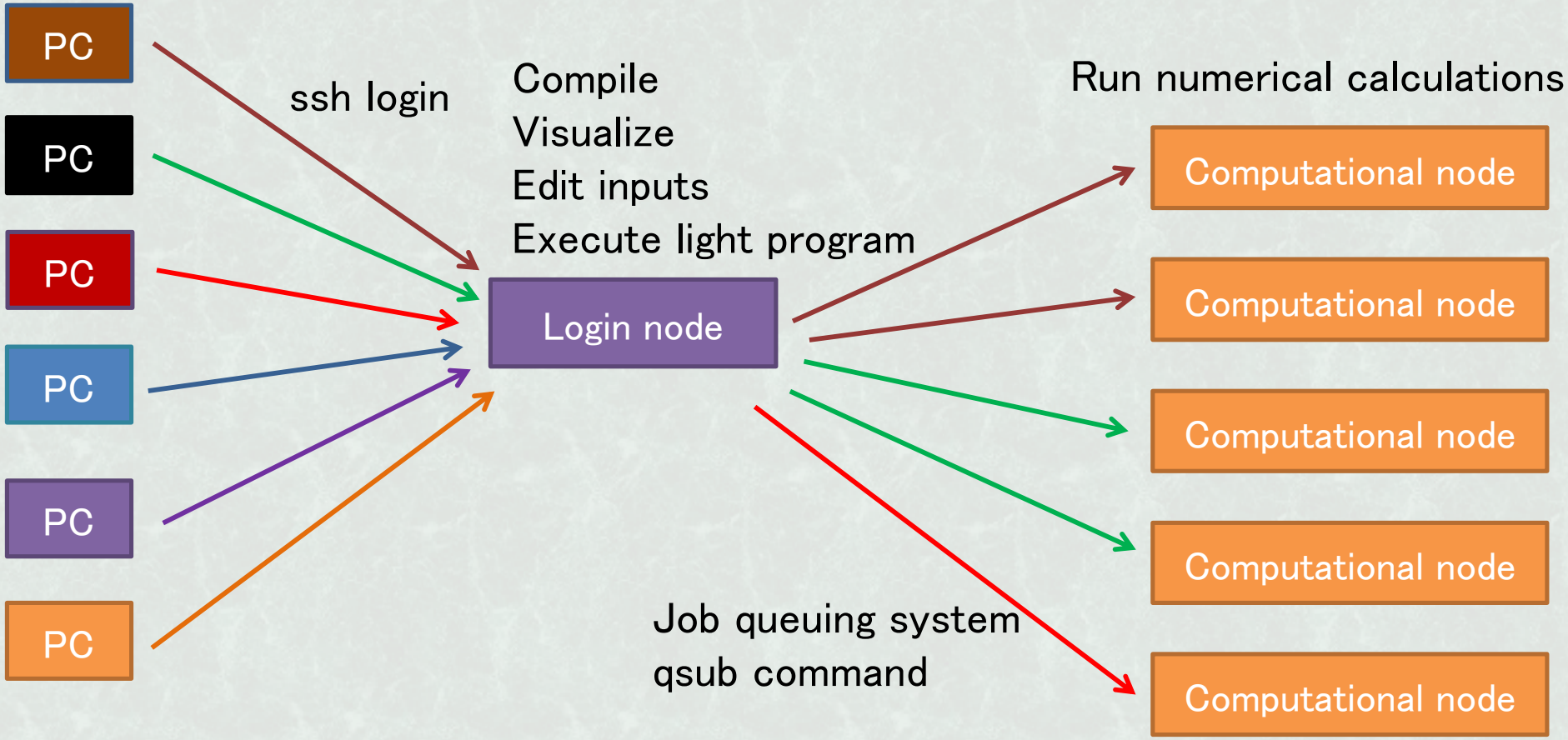
→ Plot DOS and Fermi surface

Free exercise time

Login

# Computer for this hands-on

CPU : Intel Xeon Broadwell 14cores × 2 (28 cores/node)
RAM : 64 GB / node
Compiler : Intel compiler & MKL library
Utilities : /home/public/bin/ (already added to PATH)



Login

# Login to workstation

## Case 1: Use ssh command (Linux, UNIX, macOS, WSL)

```
$ ssh -Y user-name@server-name
```

Your PC may ask you to add this workstation to “known-hosts”.

Enter your password.

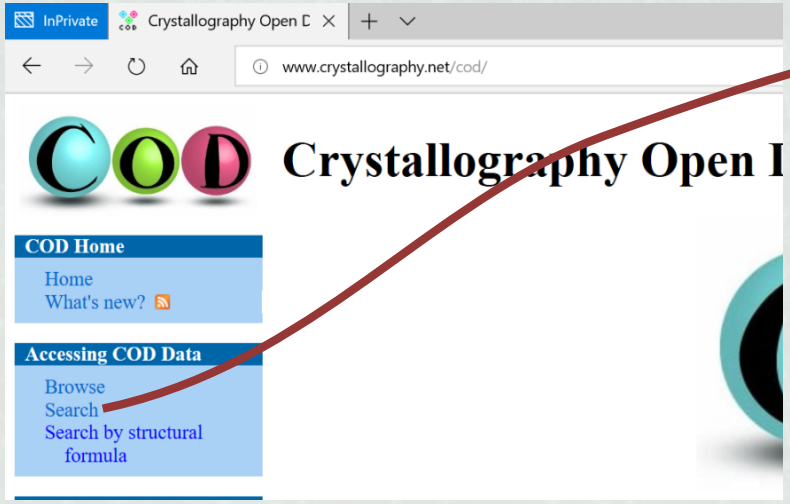
## Case 2: GUI ssh-client in Windows

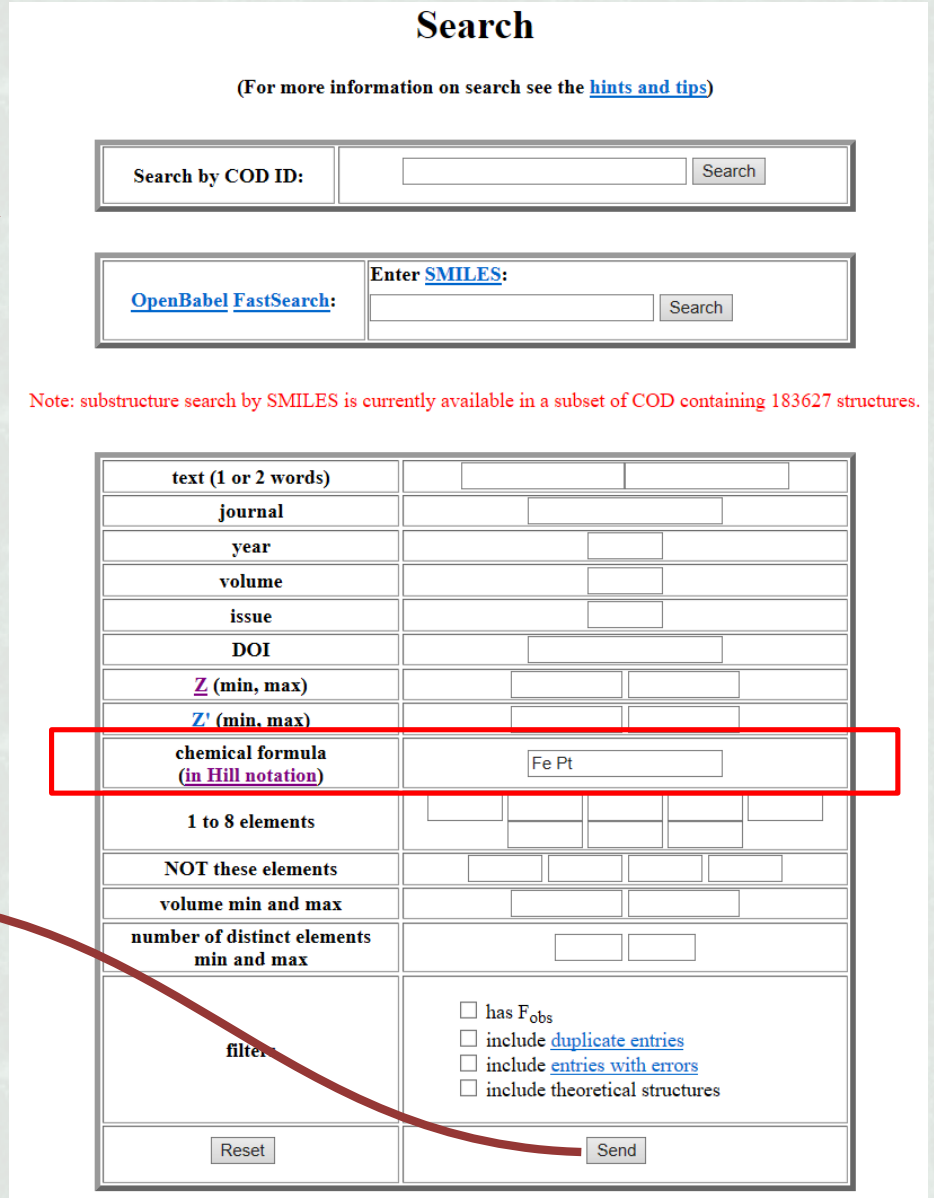
Google “xming putty”

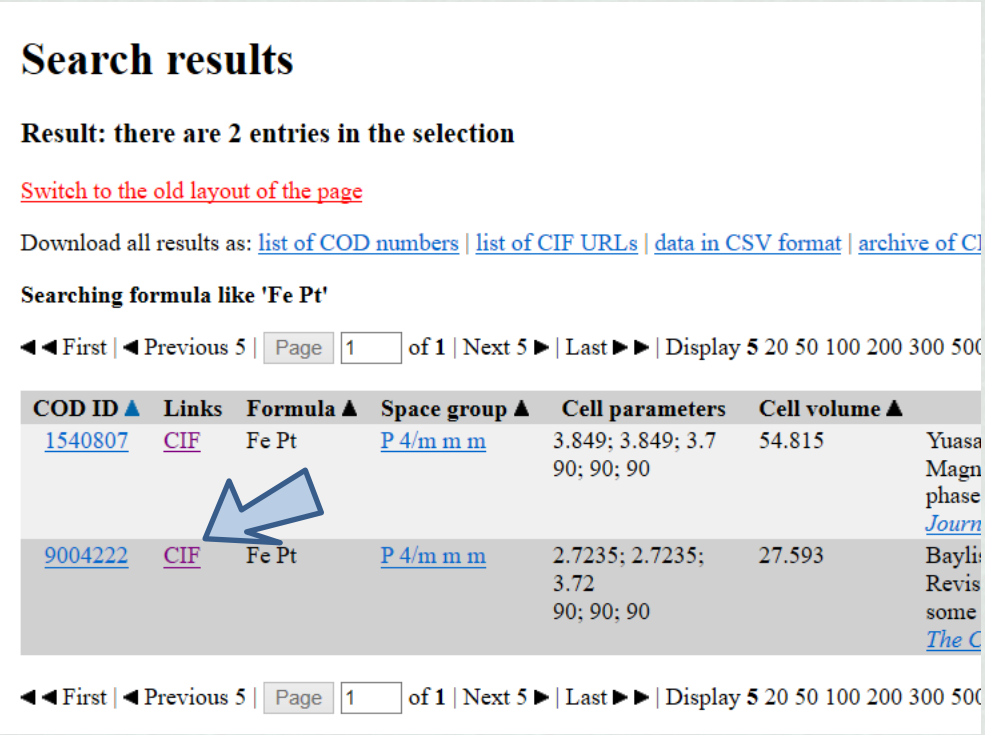

WinSCP is also required in this case.

# Get CIF file from COD

Google “Crystallography Open Database”



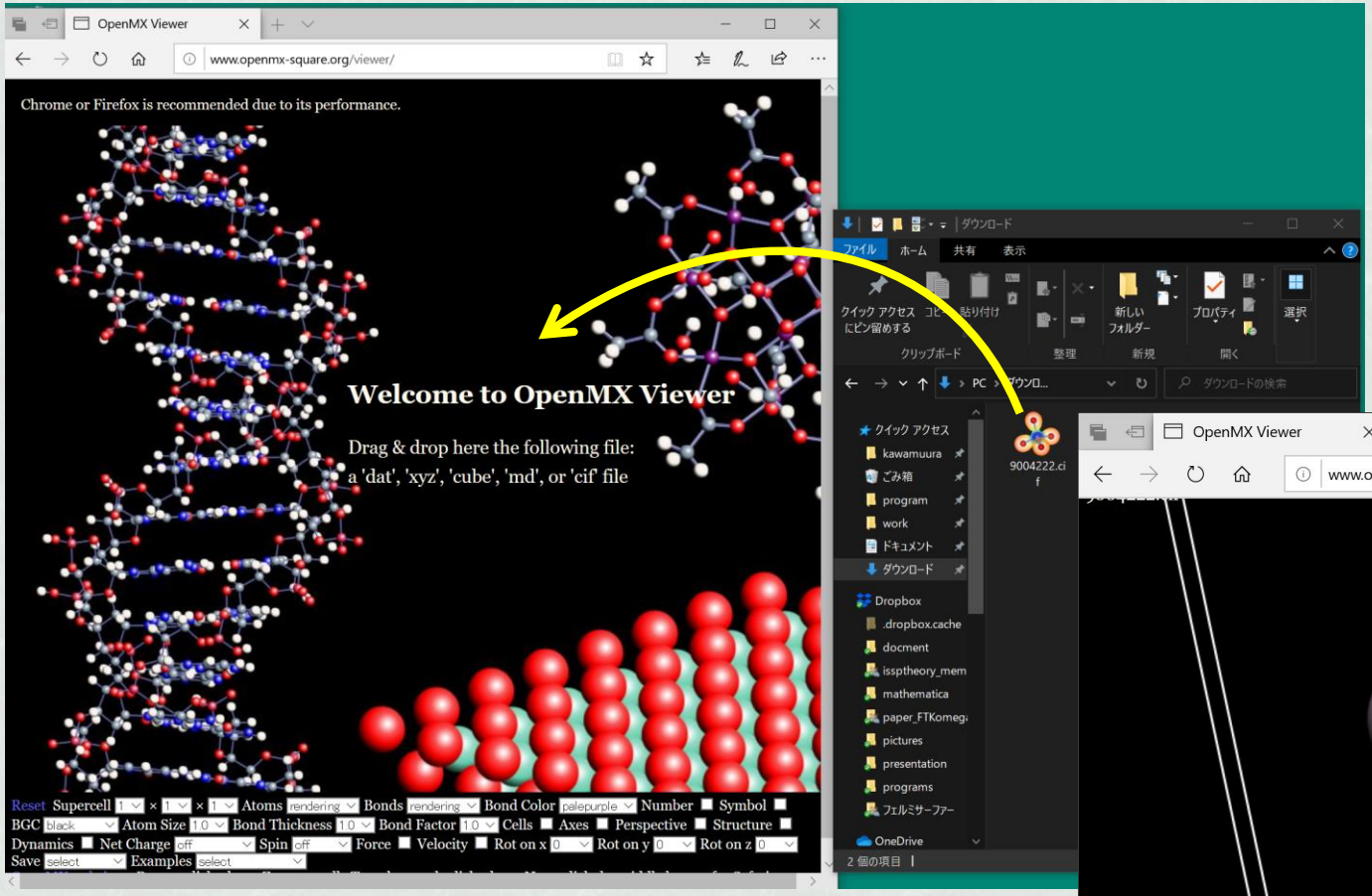




“chemical formula” in **alphabetic** order:

e.g. $Nd_2Fe_{14}B$ → ″B Fe14 Nd2″

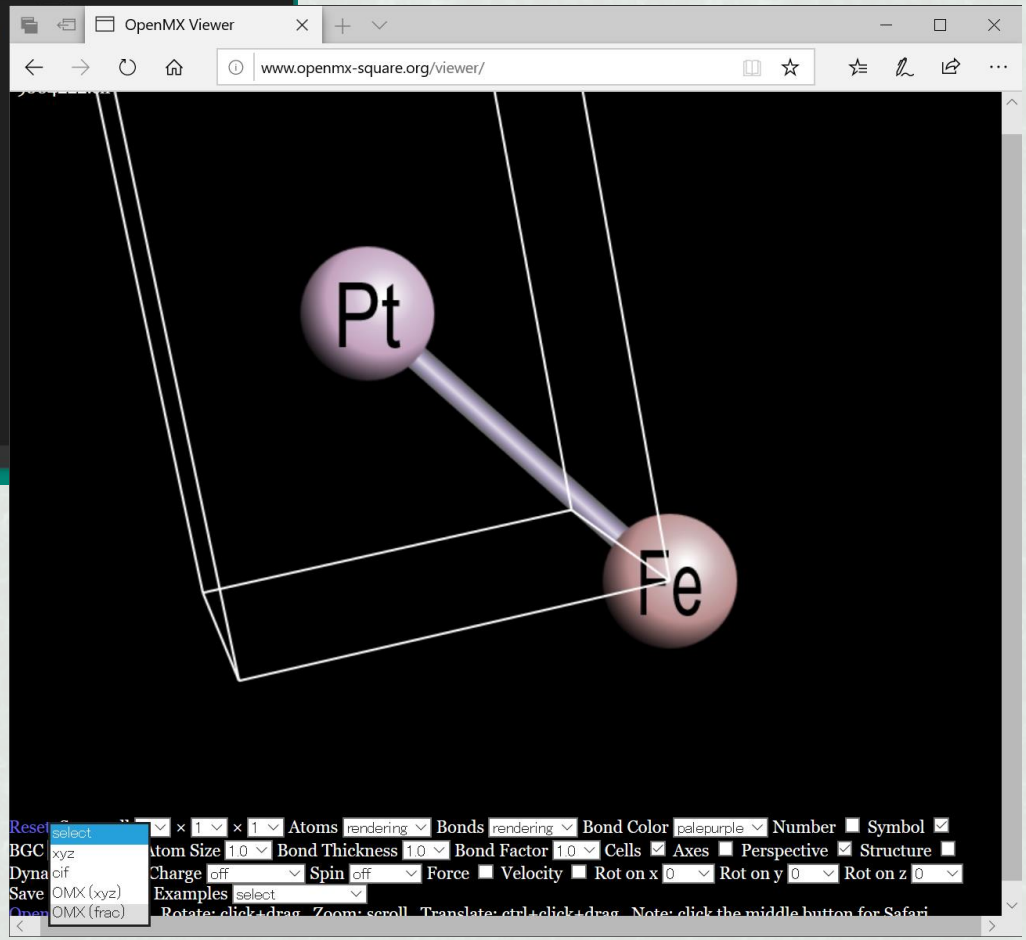
# Convert to OpenMX input

Search "OpenMX Viewer" on Web










"Save" → "OMX (frac)"

Download an OpenMX input "abc.dat"

# Send into workstation and edit

In your PC

Depends on your PC

```
$ cd ~/Download/
$ scp abc.dat gauss.issp.u-tokyo.ac.jp:~/
$ ssh -Y guest〇〇@gauss.issp.u-tokyo.ac.jp
```

In gauss, open "abc.dat" and apply the following modification:

1, Add line

```
data.path /home/public/program/openmx3.9/DFT_DATA19/
```

2, Modify k-grid

```
scf.Kgrid 6 6 5
```

```
scf.Kgrid 12 12 10
```

# Queuing system

```
$ cp /home/hands-on/practice/openmx.sh .
$ qsub openmx.sh
```

MPI processes per node

Number of OpenMP threads

```
#!/bin/sh
#PBS -l nodes=1:ppn=28
#
source ~/.bashrc
#
export OMP_NUM_THREADS=1
#
cd $PBS_O_WORKDIR
#
mpiexec -hostfile $PBS_NODEFILE /home/public/bin/openmx abc.in -nt $OMP_NUM_THREADS ¥
   > openmx.stdout
```

Job status

ppn × OMP_NUM_THREADS ≦ number of cores par node (28)

```
$ qstat -a

gauss:

                                                                  Req'd    Req'd       Elap
Job ID                  Username    Queue    Jobname          SessID  NDS   TSK    Memory   Time    S   Time
----------------------- ----------- -------- ---------------- ------ ----- ------ ------ --------- - ---------
168.gauss               kawamura    default  openmx.sh         23466     1      7    --         --  R  00:00:00
```

Nodes

R: Running

Q: Waiting to start

```
$ qdel job-ID
```

Delete job (If you submitted wrong job, etc.)

# Check standard output

```
$ less openmx.stdout
```

```
******************** MD= 1 SCF= 6 ********************
<Poisson> Poisson's equation using FFT...
<Set_Hamiltonian> Hamiltonian matrix for VNA+dVH+Vxc...
<Band> Solving the eigenvalue problem...
KGrids1: -0.45833 -0.37500 -0.29167 -0.20833 -0.12500 -0.04167 0.04167 ...
KGrids2: -0.45833 -0.37500 -0.29167 -0.20833 -0.12500 -0.04167 0.04167 ...
KGrids3: -0.45000 -0.35000 -0.25000 -0.15000 -0.05000 0.05000 0.15000 ...
<Band_DFT> Eigen, time=0.235195
<Band_DFT> DM, time=0.416772
1 Fe MulP 9.2692 6.4700 sum 15.7391 diff 2.7992
2 Pt MulP 8.2556 8.0052 sum 16.2609 diff 0.2504
Sum of MulP: up = 17.52480 down = 14.47520
total= 32.00000 ideal(neutral)= 32.00000
<DFT> Total Spin Moment (muB) = 3.049605408328
<DFT> Mixing_weight= 0.300000000000
<DFT> Uele = -36.891552072687 dUele = 0.096556004642
<DFT> NormRD = 1.170298176078 Criterion = 0.000000100000
```

Time for eigen-solution and density matrix (second)

Residual Norm

We can monitor this during the calculation.

# Check the summary of output

```
$ less abc.out
```

Each section is surrounded by "*****"
Therefpre in "less" program, we can search
/¥*¥*⏎
type "n" and "N" for next and previous match

```
**********************************
**********************************
SCF history at MD= 1
**********************************
**************************************************

SCF= 1 NormRD= 1.000000000000 Uele= -35.799412526515
SCF= 2 NormRD= 1.889891043491 Uele= -35.799412526515
SCF= 3 NormRD= 1.868362109120 Uele= -35.827036113047
SCF= 4 NormRD= 1.414230084310 Uele= -36.568305665058
SCF= 5 NormRD= 1.455962148448 Uele= -36.794996068046
SCF= 6 NormRD= 1.170298176078 Uele= -36.891552072687
SCF= 7 NormRD= 1.067138991513 Uele= -36.946866542123
SCF= 8 NormRD= 0.743951621288 Uele= -37.028028215533
```

```
**********************************
Total energy (Hartree) at MD = 1
**********************************
```

```
**********************************
Eigenvalues (Hartree) of SCF KS-eq.
**********************************
```

Other sections

- Mulliken populations
- Dipole moment (Debye)
- xyz-coordinates (Ang) and forces (Hartree/Bohr)
- etc.

Try to jump and see each section.

# Band structure

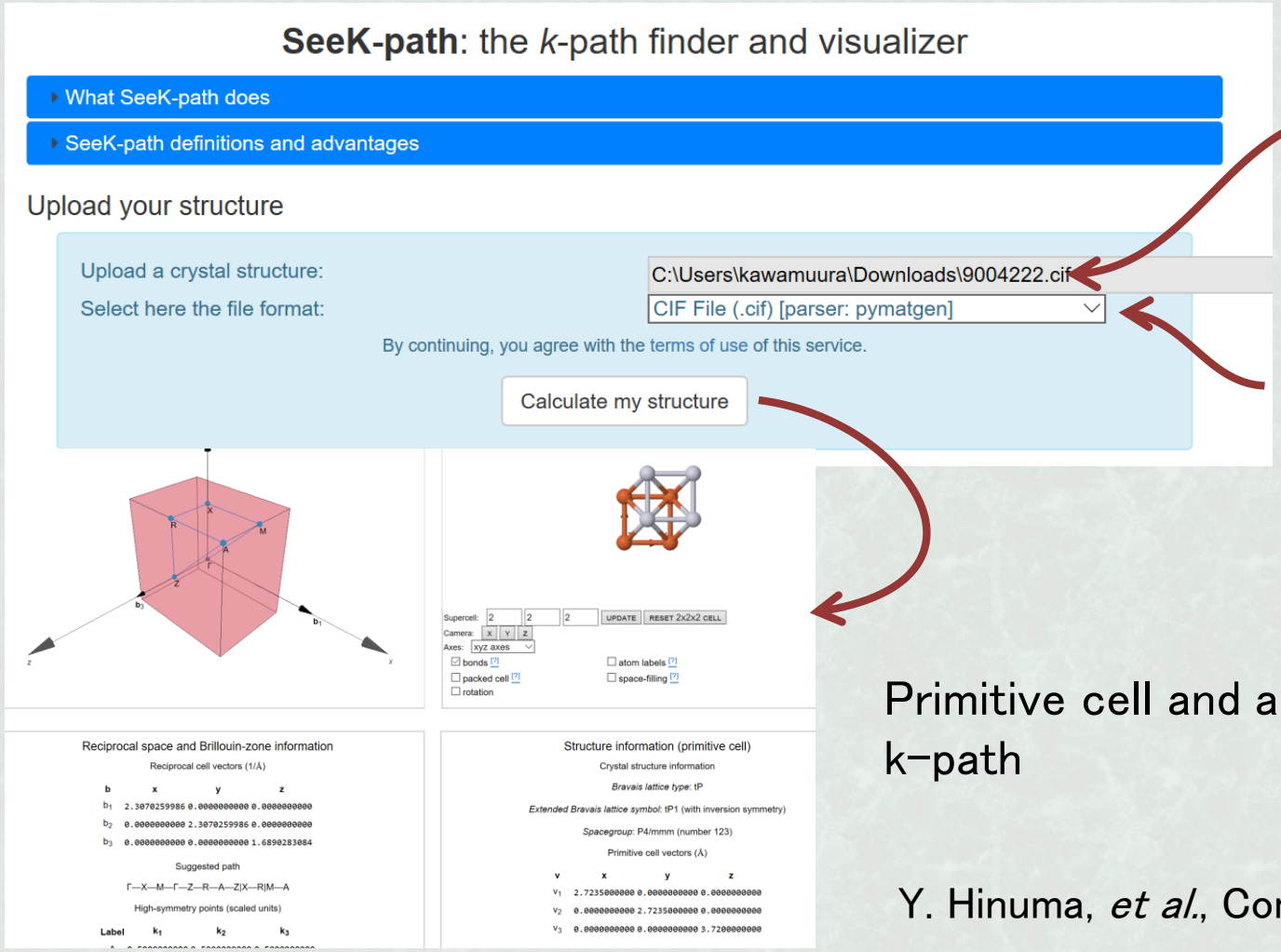
Edit “abc.dat” with emacs, vim, etc.

```
orbitalOpt.Force.Skip    on
scf.restart on

Band.dispersion on # on|off, default=off
```

Restart from the converged charge density

For the k-path serection, in the Web browser of your PC, search ″SeeK-path″




Downloaded CIF file

″CIF File (parser: pymatgen)″

Primitive cell and appropriate k-path

Y. Hinuma, *et al.*, Comp. Mat. Sci. **128**, 140 (2017).

# Specify k-path

Structure information (primitive cell)

Crystal structure information

*Bravais lattice type*: tP

*Extended Bravais lattice symbol*: tP1 (with inversion symmetry)

*Spacegroup*: P4/mmm (number 123)

Primitive cell vectors (Å)

| v | x | y | z |
|---|---|---|---|
| $v_1$ | 2.7235000000 | 0.0000000000 | 0.0000000000 |
| $v_2$ | 0.0000000000 | 2.7235000000 | 0.0000000000 |
| $v_3$ | 0.0000000000 | 0.0000000000 | 3.7200000000 |

- Quantum ESPRESSO pw.x input
- CP2K input
- CRYSTAL *.d3 input
- VASP KPOINTS input for LDA/GGA

[FORMAT DESCRIPTION]

Copy to clipboard (Replace <...> with the actual content for your simulation)

```
Special k-points for band structure
<...>  ! intersections
line-mode
reciprocal
    0.0000000000     0.0000000000     0.0000000000 1    GAMMA
    0.0000000000     0.5000000000     0.0000000000 1    X


    0.0000000000     0.5000000000     0.0000000000 1    X
    0.5000000000     0.5000000000     0.0000000000 1    M
```

```
<Band.KPath.UnitCell
2.7235000000 0.0000000000 0.0000000000
0.0000000000 2.7235000000 0.0000000000
0.0000000000 0.0000000000 3.7200000000
Band.KPath.UnitCell>
```



```
Band.Nkpath 9
<Band.kpath
15 0.0000000000 0.0000000000 0.0000000000 0.0000000000 0.5000000000 0.0000000000 GAMMA X
15 0.0000000000 0.5000000000 0.0000000000 0.5000000000 0.5000000000 0.0000000000 X M
15 0.5000000000 0.5000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000 M GAMMA
15 0.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000 0.5000000000 GAMMA Z
15 0.0000000000 0.0000000000 0.5000000000 0.0000000000 0.5000000000 0.5000000000 Z R
15 0.0000000000 0.5000000000 0.5000000000 0.5000000000 0.5000000000 0.5000000000 R A
15 0.5000000000 0.5000000000 0.5000000000 0.0000000000 0.0000000000 0.5000000000 A Z
15 0.0000000000 0.5000000000 0.0000000000 0.0000000000 0.5000000000 0.5000000000 X R
15 0.5000000000 0.5000000000 0.0000000000 0.5000000000 0.5000000000 0.5000000000 M A
Band.kpath>
```

# Band structure calculation

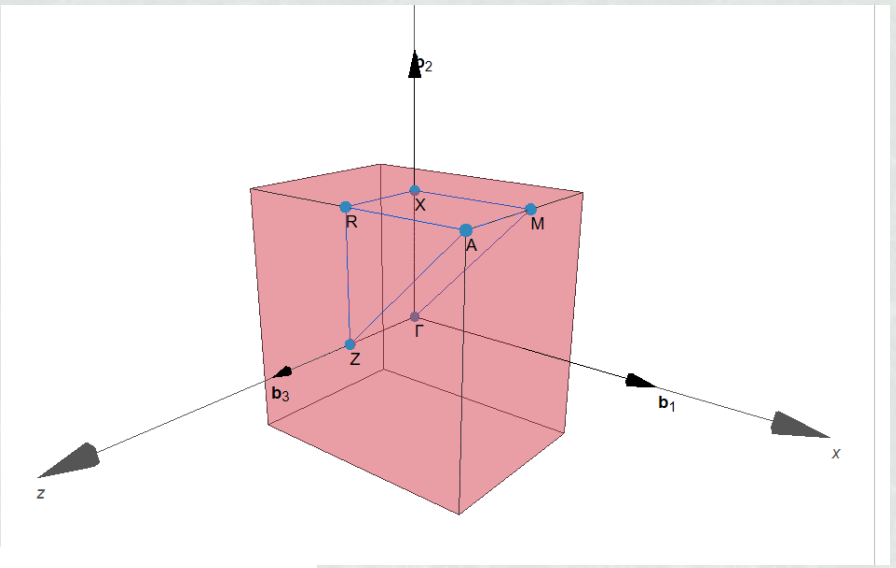
```
$ qsub openmx.sh
```

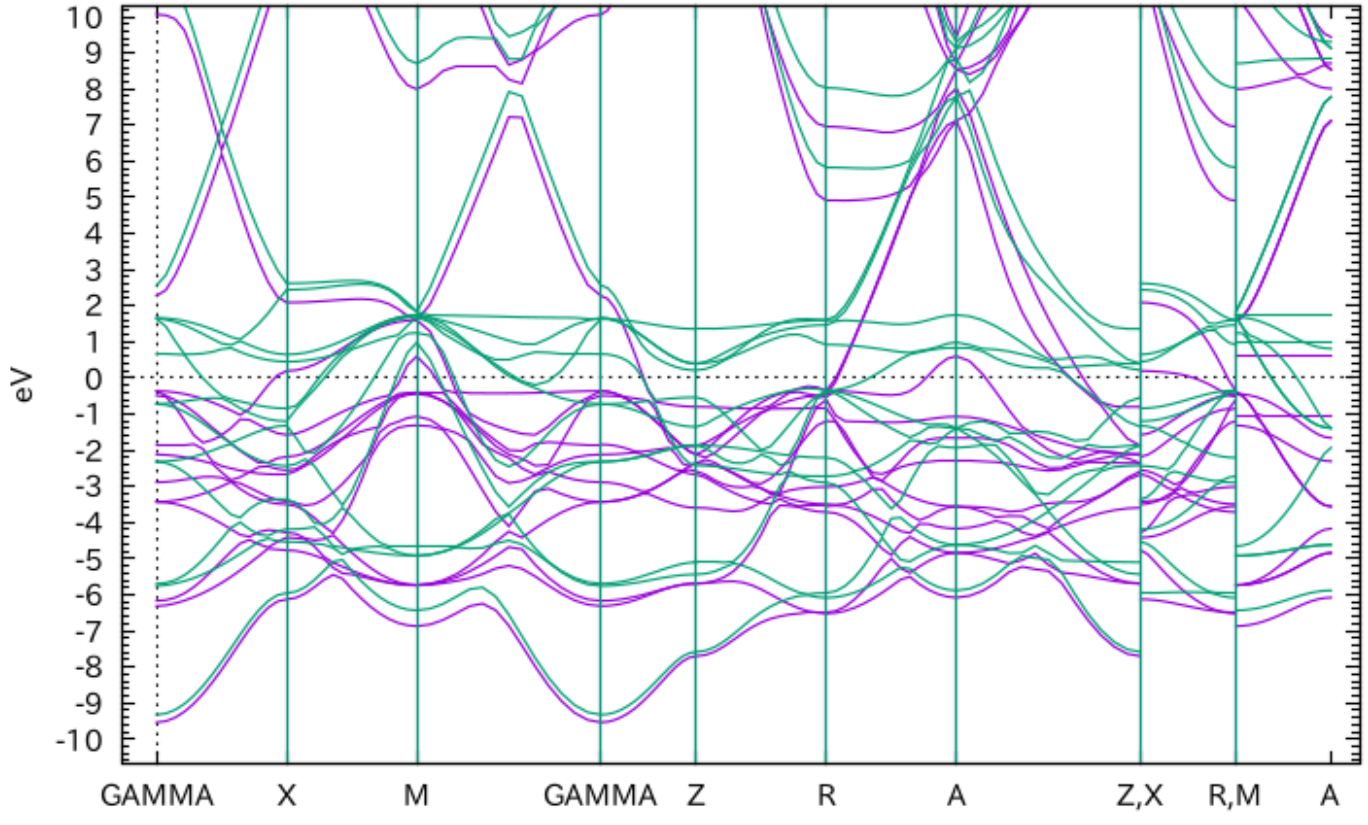
→ `abc.Band`

```
$ bandgnu13 abc.Band
```

→ `abc.GNUBAND`

```
$ gnuplot abc.GNUBAND
```



# Kohn-Sham orbitals

Modify abc.dat

```
Band.dispersion    off
```

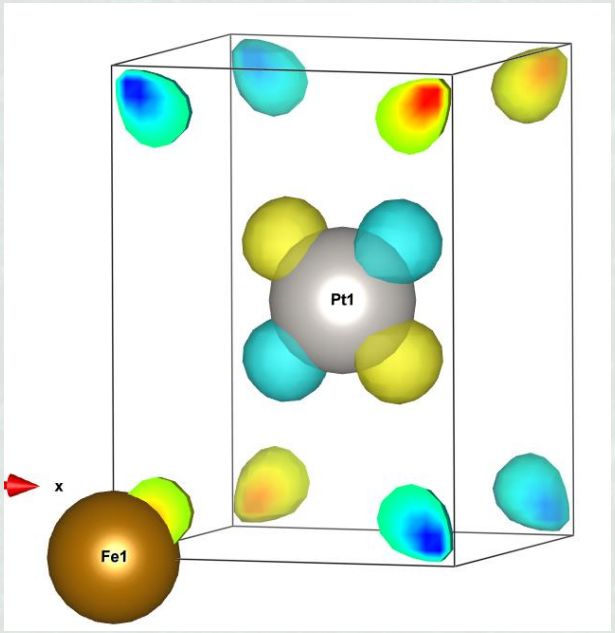
Add:

```
MO.fileout      on
num.HOMOs   1
num.LUMOs   2
MO.Nkpoint   1
<MO.kpoint
0.0   0.0   0.0
MO.kpoint>
```

```
$ qsub openmx.sh
```

```
abc.homo0_0_0_i.cube   abc.lumo0_0_0_i.cube   abc.lumo0_0_1_i.cube
abc.homo0_0_0_r.cube   abc.lumo0_0_0_r.cube   abc.lumo0_0_1_r.cube
```

```
$ VESTA *_r.cube
```



# Fermi surface, DOS, PDOS (1)

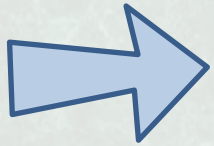
Modify abc.dat

```
MO.fileout      off
```

Add:

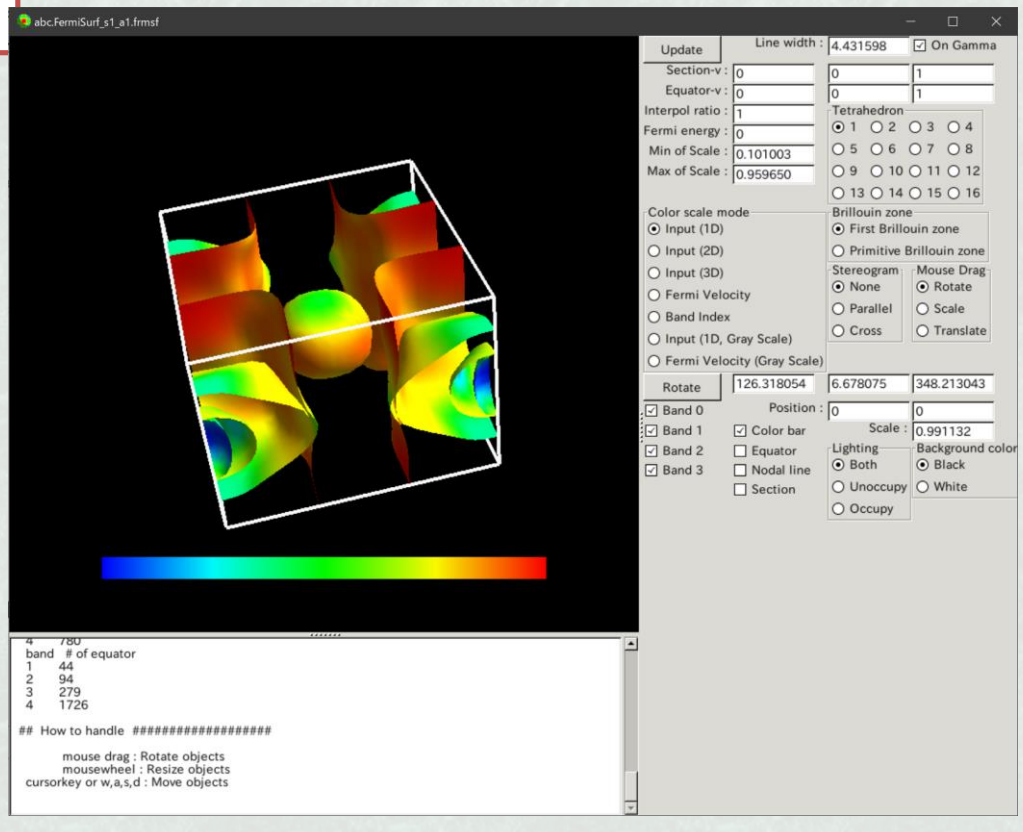
```
Dos.fileout on
Dos.Erange -15.0 10.0
Dos.Kgrid 24 24 20
FermiSurfer.fileout on
```

```
$ qsub openmx.sh
```



```
abc.Dos.val, abc.Dos.vec,
abc.FermiSurf_s0_a1.frmsf,
abc.FermiSurf_s0_a2.frmsf,
abc.FermiSurf_s1_a1.frmsf,
abc.FermiSurf_s1_a2.frmsf
```

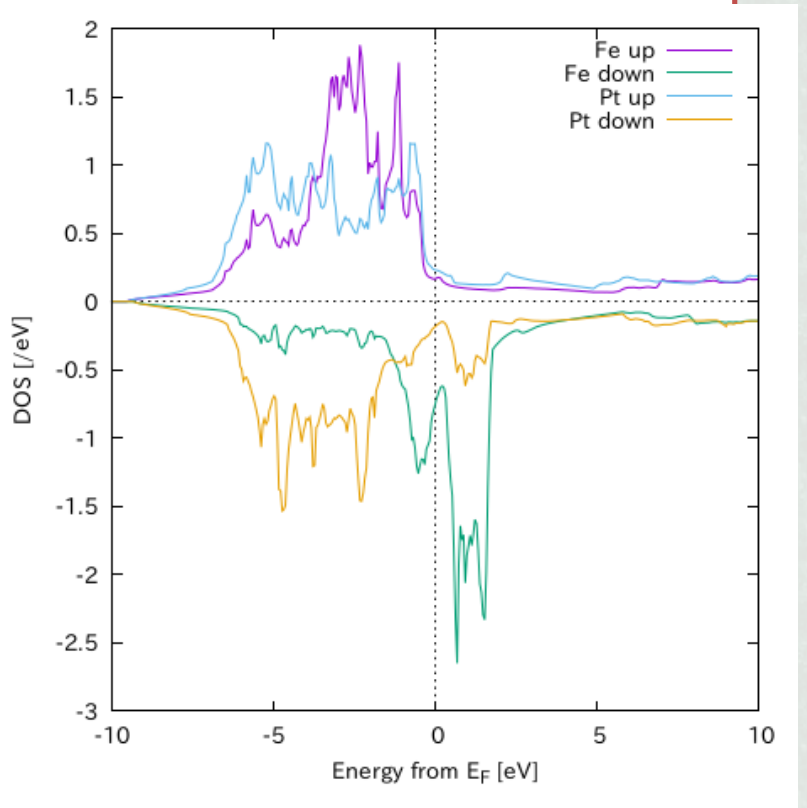
```
$ fermisurfer abc.FermiSurf_s1_a1.frmsf
```



# Fermi surface, DOS, PDOS (2)

*Interactive queue*

```
[XXX@gauss] $ qsub -I
[XXX@gaussXX] $ DosMain abc.Dos.val abc.Dos.vec
:
Which method do you use?, Tetrahedron(1), Gaussian Broadening(2)
1 ⏎
Do you want Dos(1) or PDos(2)?
2 ⏎
Which atoms for PDOS : (1,...,3), ex 1 2
1 2 ⏎
[XXX@gaussXX] $ exit
$ gnuplot
gnuplot> set xzeroaxis
gnuplot> set yzeroaxis
gnuplot> set xlabel "Energy from E_F [eV]"
gnuplot> set ylabel "DOS [/eV]"
gnuplot> plot [-10:10] ¥
"abc.PDOS.Tetrahedron.atom1" u 1:2 w l t "Fe up"
"abc.PDOS.Tetrahedron.atom1" u 1:3 w l t "Fe dow
"abc.PDOS.Tetrahedron.atom2" u 1:2 w l t "Pt up"
"abc.PDOS.Tetrahedron.atom2" u 1:3 w l t "Pt dow
```



# Free exercise time

- Compute other materials from COD database. Explain the result of band structure, PDOS, Fermisurface.
- Install the following visualization tools into your own PC.
  - VESTA
  - FermiSurfer
- Try other functions of OpenMX (Wannier function, etc.).
- Etc.