

# 4. Utilities & Summary

Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

NAIIS-ENCCS training, online 17-18<sup>th</sup> Apr 2024

# VASP - Best Practices Workshop



<https://www.nsc.liu.se/>

**NAISS**

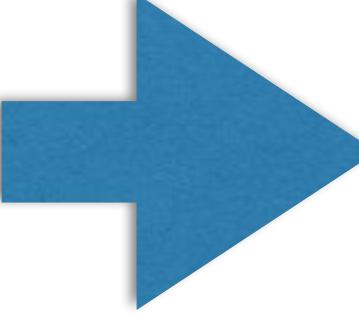
<https://www.naiss.se/>



EuroCC National Competence Centre Sweden

<https://enccs.se/>

# Utilities & Tools

- Commercial (a few) / free to use (many)
- Different types of licenses (check!)
- Install on Linux/Unix systems (mostly)  
... your own Linux or Mac computer
- Non-commercial  used at NSC clusters

...the selection reflects the tools VASP users are using at NSC clusters

... clickable links are underlined



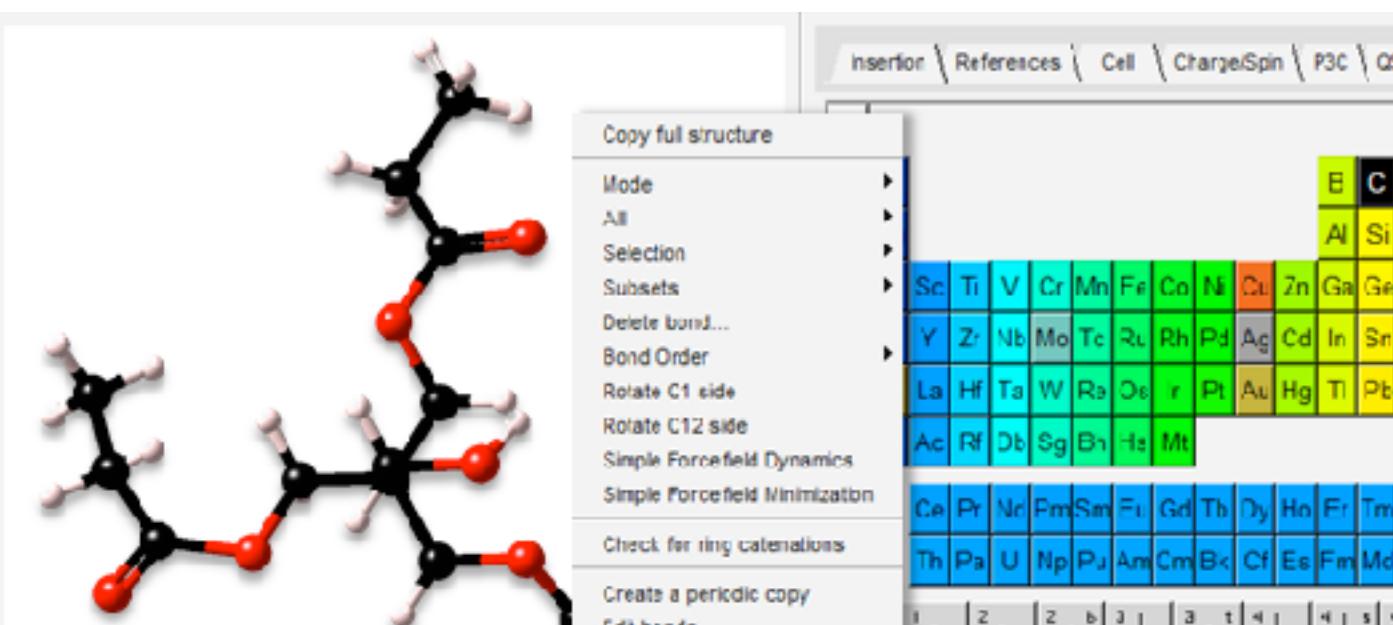
# MedeA

- MedeA VASP
- Collaboration with VASP developers
- Lots of different tools
- Interesting seminars!

Wednesday's Webinar Session

## The Random Phase Approximation: A Practical Method Beyond DFT

-> Commercial software



<https://www.materialsdesign.com/>

### MedeA Standard Builders

- *MedeA Crystal Builder*
- *MedeA Molecular Builder*
- *MedeA Conformer Search*
- *MedeA Supercell Builder*
- *MedeA Special Quasirandom Structures (SQS)*
- *MedeA Polymer Builder*
- *MedeA Nanobuilder*
- *MedeA Surface Builder*
- *MedeA Stack Layer Builder*
- *MedeA Substitutional Builder*
- *MedeA Random Substitutions*

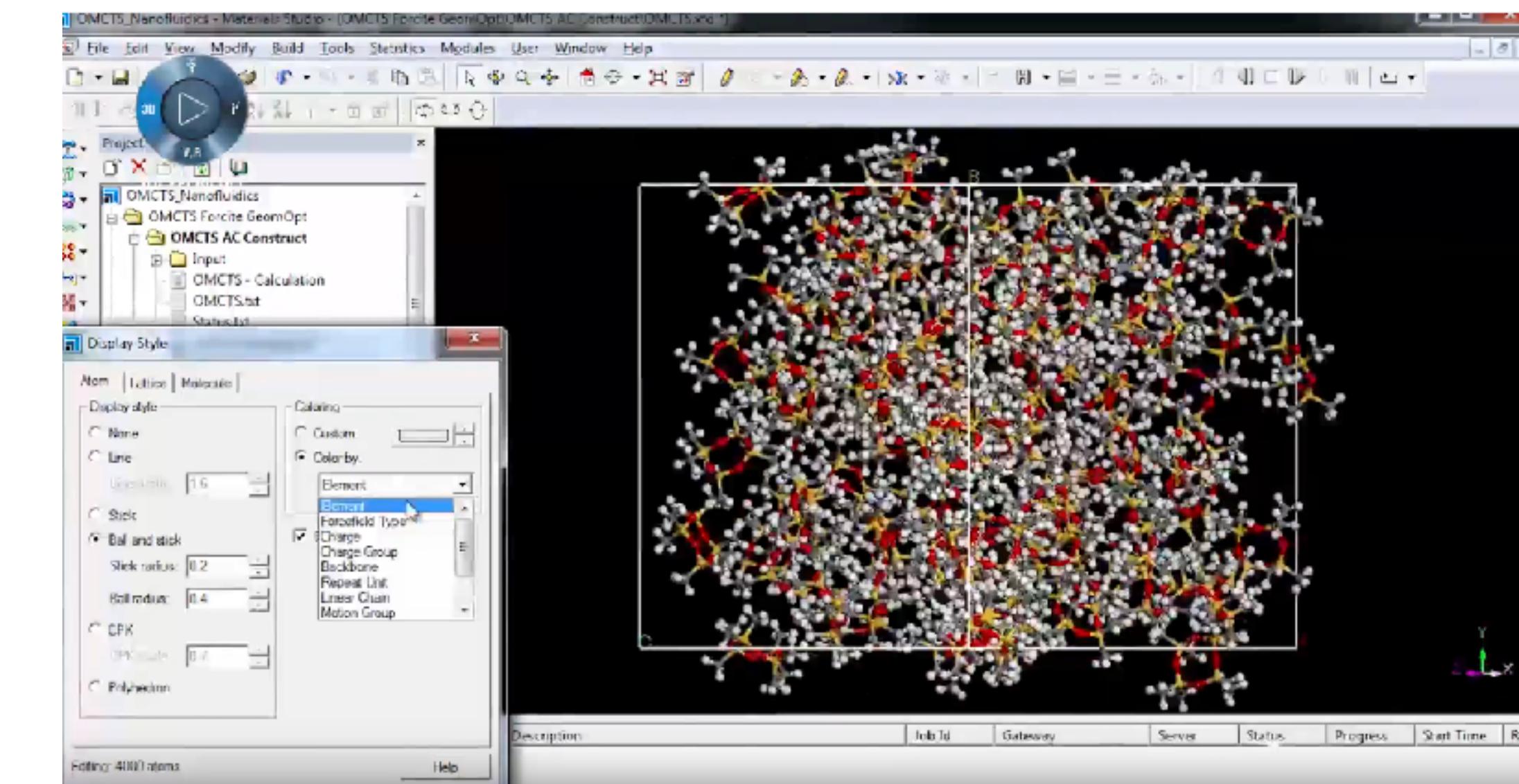
### MedeA Specialty Builders

- *MedeA Amorphous Materials Builder*
- *MedeA Thermoset Builder*
- *MedeA Interface Builder*
- *MedeA Mesoscale Builder*

# Materials Studio



- Good builder bulk/interfaces etc.
- Part of a larger package, integrated with CASTEP, GULP etc.



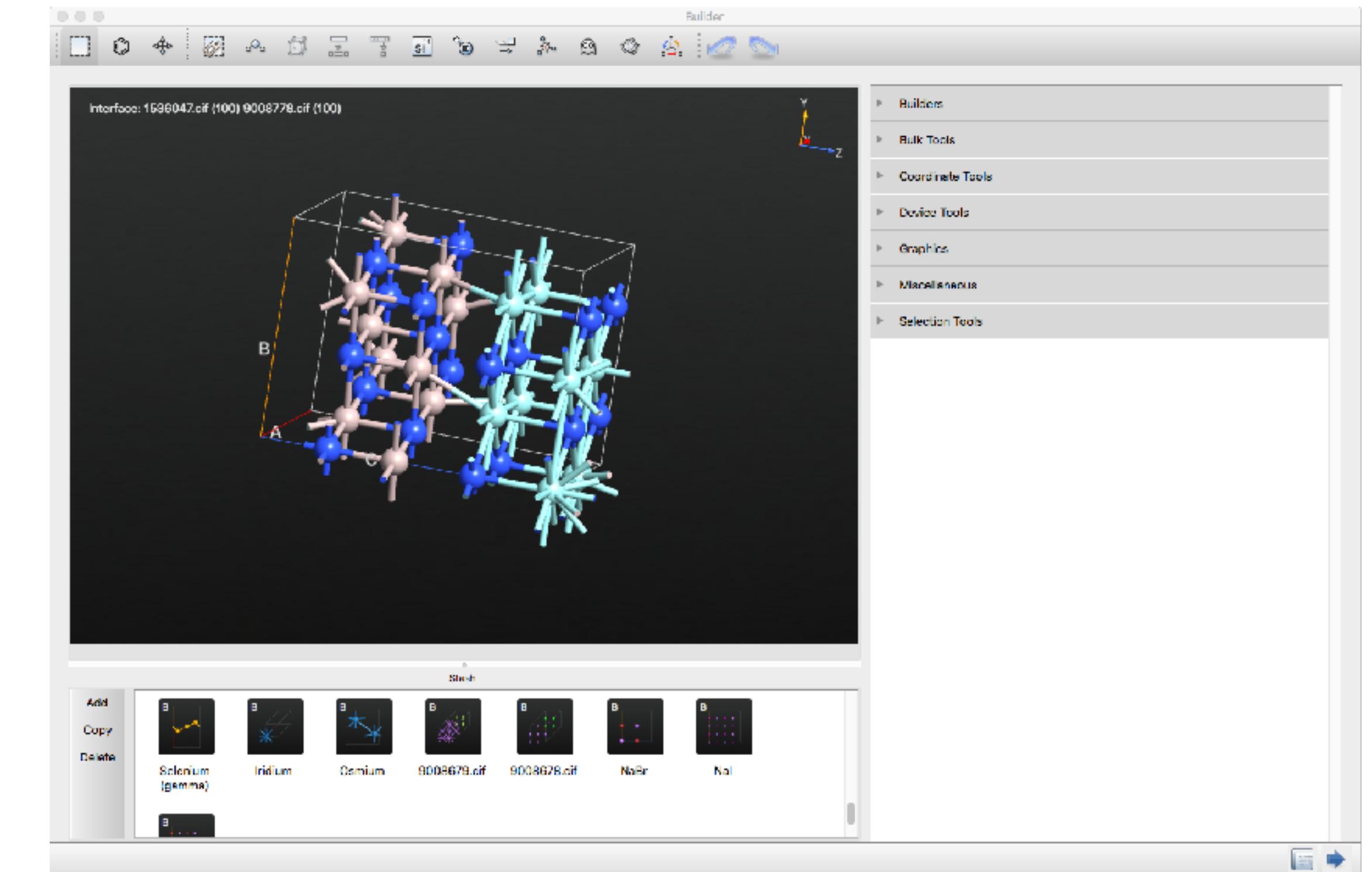
-> **Commercial software**

<https://www.3ds.com/products-services/biovia/products/molecular-modeling-simulation/biovia-materials-studio/>

# NanoLab

**SYNOPSYS®**

- Good builder bulk/interfaces etc.
- Lots of functionality
- Now part of a larger package,  
QuantumATK



*Previously: free academic lic. was available for “VirtualNanoLab”*

**-> Commercial software**

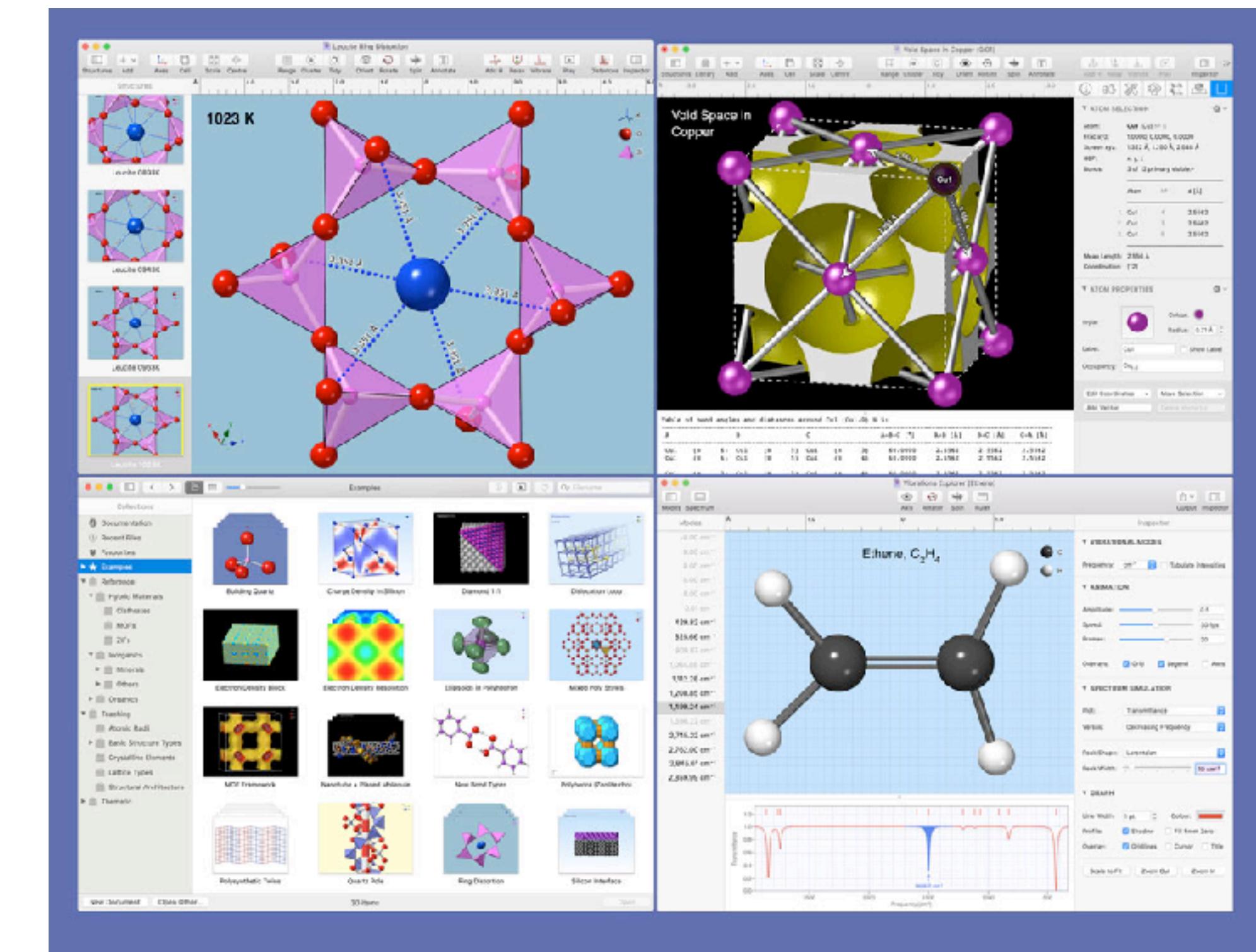
<https://www.synopsys.com/silicon/quantumatk/resources/feature-list.html#nanolab>

Quantum  
Wise

# CrystalMaker



- Good builder bulk/interfaces etc.
- Lots of functionality
- Mac/Windows



-> Commercial software

<https://crystalmaker.com/>

# Wannier90

## Calculation of Maximally-Localised Wannier Functions

- Wannier localisation scheme of Marzari and Vanderbilt [[REF](#)]
- Disentanglement scheme of Souza, Marzari and Vanderbilt [[REF](#)] for entangled bands (e.g. metals, conduction states)
- Optimised algorithm for Gamma-point calculations [[REF](#)]
- Symmetry-adapted Wannier functions [[REF](#)]
- Wannier functions without the need to define initial projections (via the SCDM method) [[REF1](#), [REF2](#)]
- Projection-only Wannier functions (without disentanglement and/or Wannierisation)
- Hamiltonian and position operators represented in the real-space Wannier function basis (eg, for use in tight-binding calculations)
- Spinor Wannier functions
- Export of Wannier functions for plotting as xsf (XCrySDen), cube format, and ray-tracing using POV-Ray
- Calculation of van der Waals energies [[REF1](#)], [[REF2](#)]
- Disentanglement within selected regions of k-space

-> Also see examples in the [VASP wiki](#)

## Transition State Tools for VASP

This is not the official VASP web page. Designed to simulate the properties of systems at the atomic scale, [VASP](#) (Vienna Ab-initio Simulation Package) is a software package created, distributed, and maintained by the Hafner Research Group at the University of Vienna.

### Vasp TST Tools

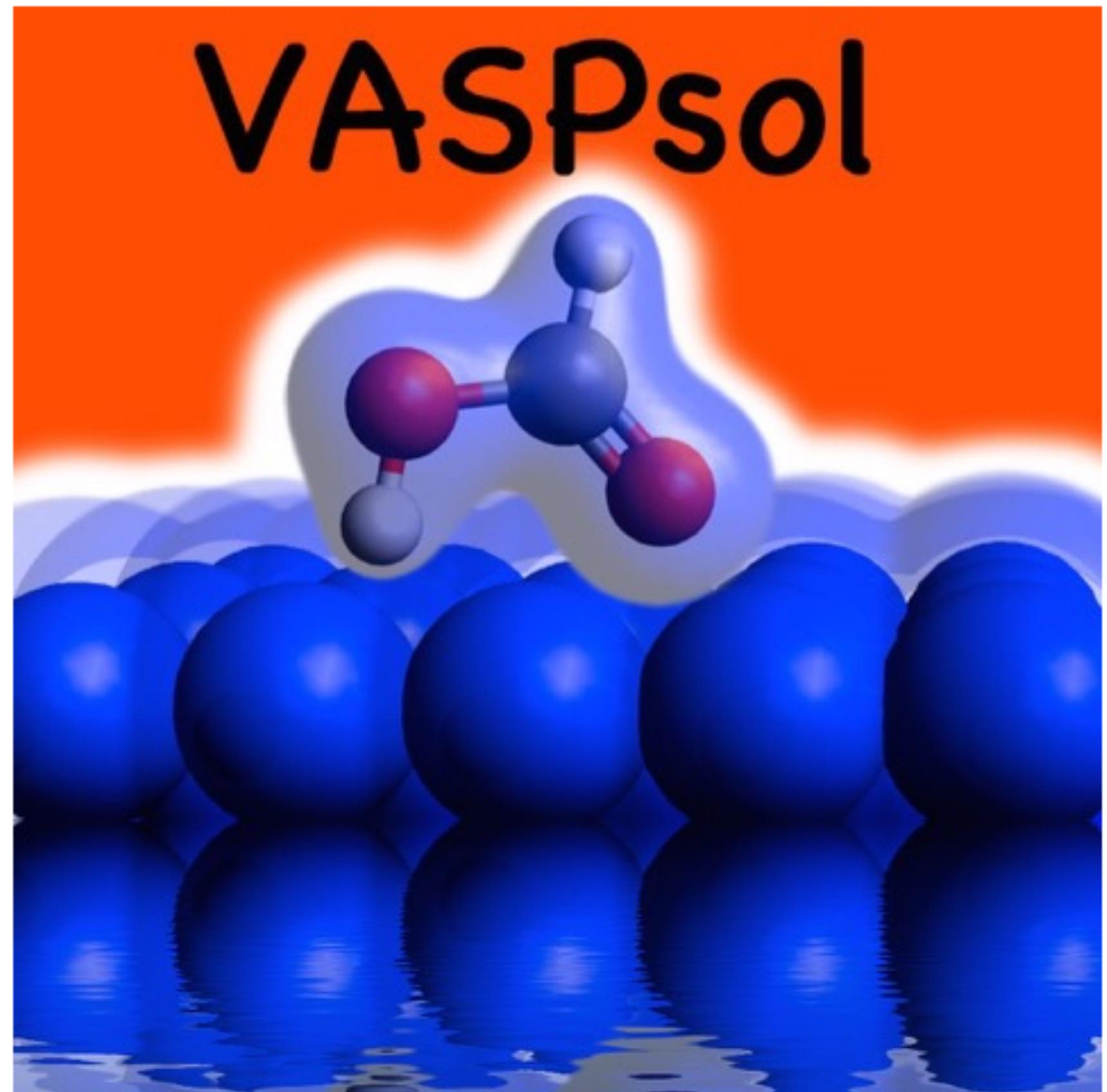
This page contains source code and scripts for finding saddle points and evaluating transition state theory (TST) rate constants with VASP.

Three saddle point finding methods and a couple other tools have been implemented to work with the VASP code. These methods are the

- [Nudged Elastic Band](#): method for finding reaction pathways between two stable states.
- [Dimer](#): method for finding reaction pathways when only one state is known.
- [Lanczos](#): provides an alternative way to find the lowest mode and find saddle points.
- [Optimizers](#): provides an alternative way to find the lowest mode and find saddle points.
- [Dynamical Matrix](#): uses finite difference to find normal modes and reaction prefactors.

**Henkelman****Group**

Also other codes, see link



# VASPsol

We have implemented an implicit solvation model that describes the effect of electrostatics, cavitation, and dispersion on the interaction between a solute and solvent into the plane-wave DFT code VASP. Our implementation provides a computationally efficient means to calculate the effects of solvation on molecules and crystal surfaces as well as reaction barriers. The strength of our solvation model implementation is its capability to handle large periodic systems such as metal and semiconductor surfaces and its interoperability with standard ultrasoft pseudopotential and projector-augmented wave potential libraries.

<https://github.com/henniggroup/VASPsol>

-> NSC VASP module

# Beef

## Bayesian Error Estimation Functionals

Electronic structure calculations with the Bayesian Error Estimation Functional (BEEF) can be performed using several DFT codes as described below. Error estimates can be obtained with the help of the [Atomic Simulation environment \(ASE\)](#) or using the stand-alone solution in libbeef. The latest version of libbeef can be obtained via:

```
svn co svn://suncatls1.slac.stanford.edu/beef/trunk beef
```

and installation is performed using:

```
./configure [options]  
make  
make install
```

# Universal Structure Predictor: Evolutionary Xtallography

# USPEX

## WHAT IS USPEX?

USPEX (Universal Structure Predictor: Evolutionary Xtallography...and in Russian "uspekh" means "success" - owing to the high success rate and many useful results produced by this method) is a method developed by the Oganov laboratory since 2004. The problem of crystal structure prediction is very old and does, in fact, constitute the central problem of theoretical crystal chemistry. In 1988 John Maddox wrote that:

*"One of the continuing scandals in the physical sciences is that it remains in general impossible to predict the structure of even the simplest crystalline solids from a knowledge of their chemical composition solids such as crystalline water (ice) are still thought to lie beyond mortals' ken".*

USPEX code solves this problem and allows to predict crystal structure with arbitrary P-T conditions by knowing only chemical composition of the material. Nowdays, it is used by over 6000 researchers worldwide. The First Blind Test of Inorganic Crystal Structure Prediction shows that USPEX outperforms other methods in terms of efficiency and reliability. The method continues to be rapidly developed.

In addition to crystal structure prediction, USPEX can work in other dimensionalities and predict the structure of nanoparticles, polymers, surfaces, interfaces and 2D-crystals. It can very efficiently handle molecular crystals (including those with flexible and very complex molecules). Moreover, it can predict stable chemical compositions and corresponding crystal structures, given just the names of the chemical elements. In addition to this fully non-empirical search, USPEX allows one to predict also a large set of robust metastable structures and perform several types of simulations using various degrees of prior knowledge.

USPEX can also be used for finding low-energy metastable phases, as well as stable structures of nanoparticles, surface reconstructions, molecular packings in organic crystals, and for searching for materials with desired physical (mechanical, electronic) properties. The USPEX code is based on an efficient evolutionary algorithm developed by A.R. Oganov's group, but also has options for using alternative methods (random sampling, metadynamics, corrected particle swarm optimization algorithms). USPEX is interfaced with many ab initio codes, such as VASP, SIESTA, GULP, Quantum Espresso, CP2K, CASTEP, LAMMPS, and so on.



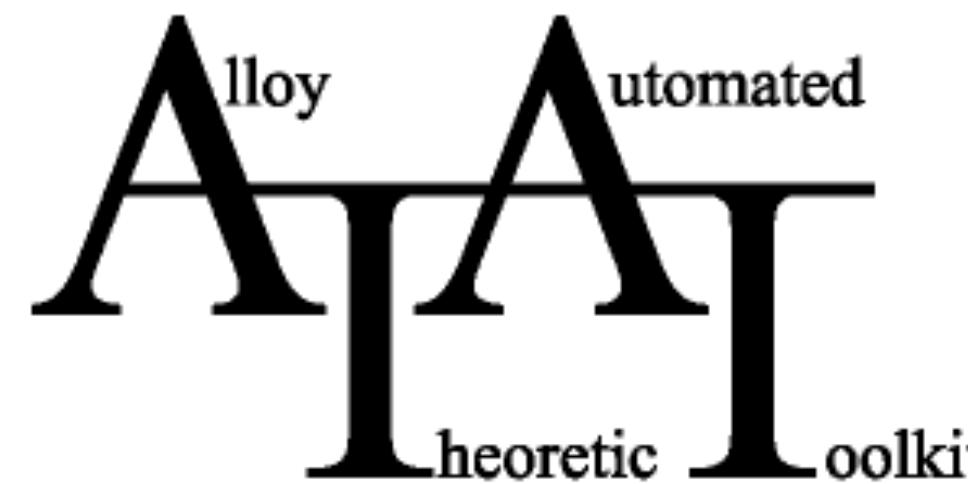
<https://uspex-team.org/en>

<https://www.nsc.liu.se/software/installed/tetralith/uspex/>

- > Apply for a personal license
- > Special instructions at NSC

# ATAT

## Alloy Theoretic Automated Toolkit (ATAT) Home Page



ATAT is a generic name that refers to a collection of alloy theory tools developed by [Axel van de Walle](#), in collaboration with various [research groups](#) and with various sources of [financial support](#).



This work is licensed under a [Creative Commons Attribution-NoDerivatives 4.0 International License](#). More details regarding how the generic license terms applies in specific context of ATAT can be found [here](#).

New!

- [ATAT forum](#)
- [The "P4" \(Piecewise Polynomial Potential Partitioning\) software for modeling mechanically unstable dynamically stabilized phases](#)
- [The Thermodynamic DataBase DataBase \(TDBDB\)](#): A search engine for openly available TDB (Thermodynamic DataBase) files.
- New [software tool for high-throughput CALPHAD](#) (CALculation of PHAse Diagram) from first-principles: `sqs2tdb` (included in ATAT).
- [New code for efficiently calculating melting points with ab initio methods: Solid Liquid in Ultra-Small Coexistence with Hovering Interface \(SLUSCHI\)](#)
- [New accelerated molecular dynamics code: Stochastic Iterations to Strengthen Yield of Path Hopping over Upper States \(SISYPHUS\)](#)

ATAT includes

- The [MIT Ab-initio Phase Stability](#) code, developed in collaboration with [Gerd Ceder's group](#), now at the University of California, Berkeley.
- The Eazy Monte Carlo Code, developed in collaboration with [Mark Asta's group](#), now at the University of California, Berkeley.
- The Constituent Strain Extension, developed in collaboration with [Alex Zunger's group](#), now at the University of Colorado, Boulder.

# Phonopy

Phonopy 2.8 Site ▾ Page ▾ Installation »

## Welcome to phonopy

Phonopy is an open source package for phonon calculations at harmonic and quasi-harmonic levels.

Phono3py is another open source package for phonon-phonon interaction and lattice thermal conductivity calculations.

See the documentation at <http://phonopy.github.io/phono3py/>

**Phonon database:** A collection of first principles phonon calculations is available as open data at

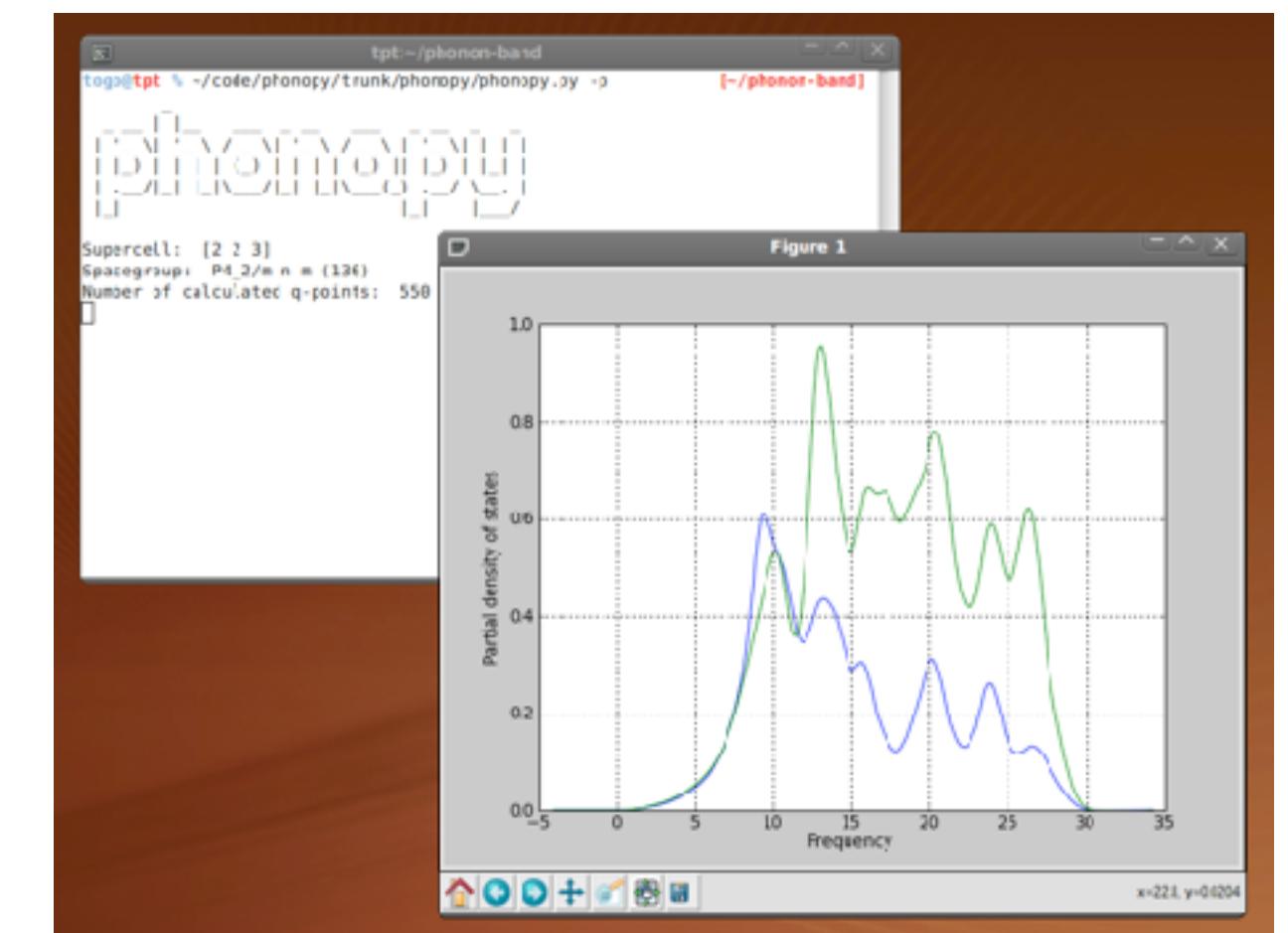
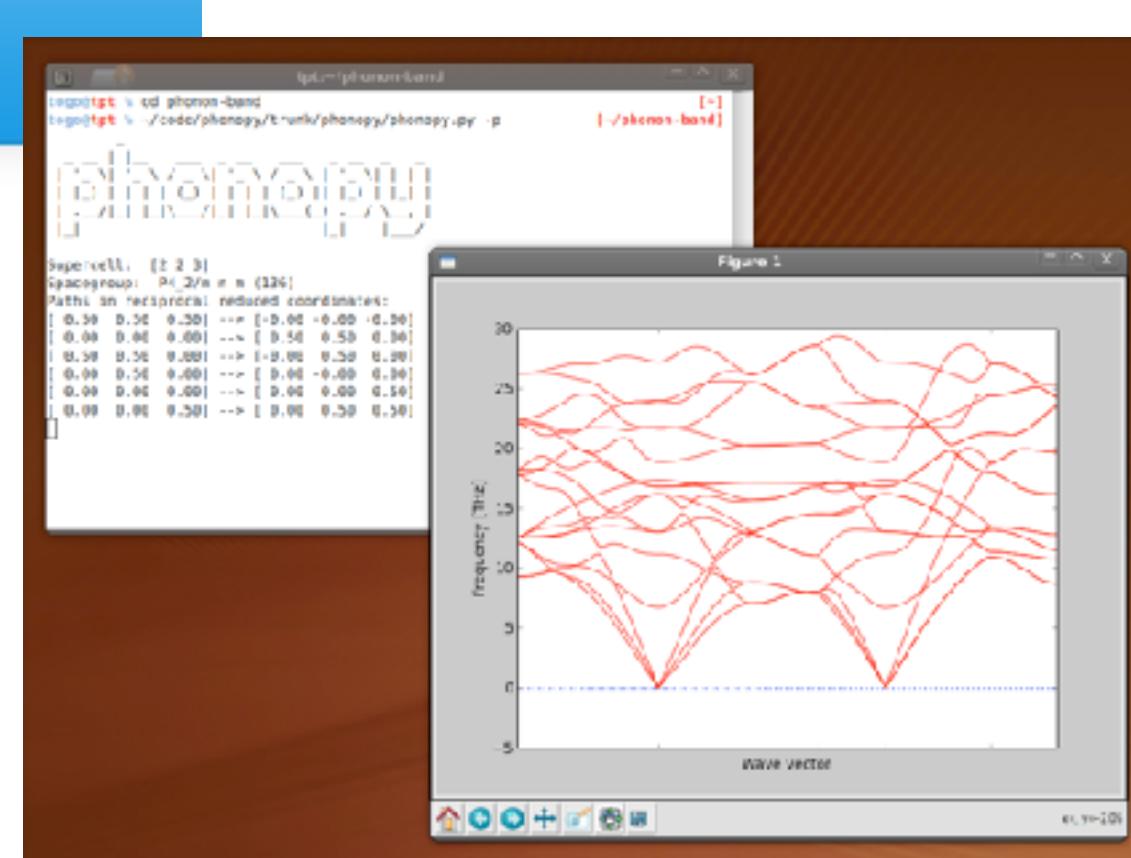
<http://phonondb.mtl.kyoto-u.ac.jp/>, where the raw data of phonopy & VASP results are downloaded.

The following features of phonopy are highlighted:

- Phonon band structure, phonon DOS and partial-DOS
- Phonon thermal properties: Free energy, heat capacity ( $C_v$ ), and entropy
- Phonon group velocity
- Thermal ellipsoids / Mean square displacements
- Irreducible representations of normal modes
- Dynamic structure factor for INS and IXS
- Non-analytical-term correction: LO-TO splitting (Born effective charges and dielectric constant are required.)
- Mode Grüneisen parameters
- Quasi-harmonic approximation: Thermal expansion, heat capacity at constant pressure ( $C_p$ )
- Interfaces to calculators: VASP, VASP DFPT, ABINIT, Quantu ESPRESSO, SIESTA, Elk, WIEN2k, CRYSTAL, DFTB+, TURBOMOLE, CP2K, FHI-aims, CASTEP, LAMMPS (external)
- Phonopy API for Python

<https://phonopy.github.io/phonopy/>

-> Install at NSC using conda (recommended)



# p4vasp

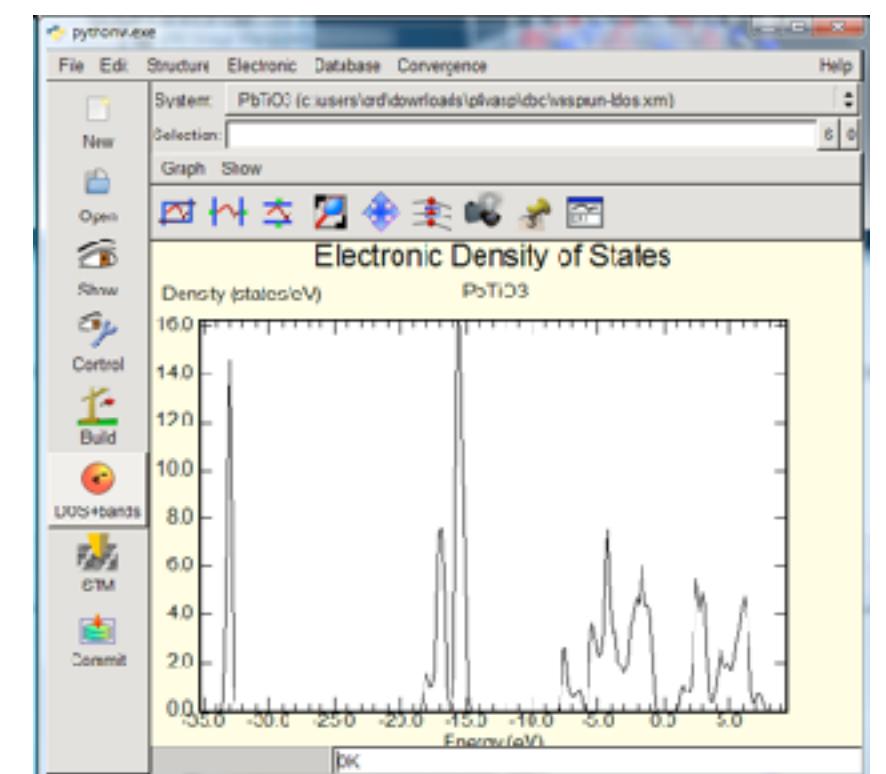
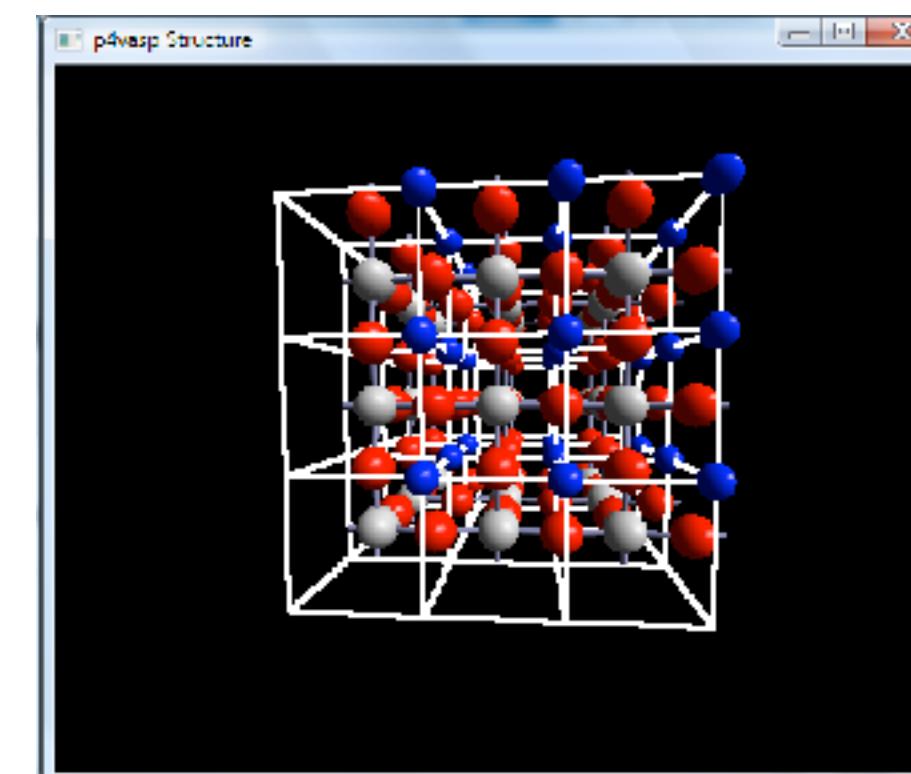
## p4vasp, the VASP Visualization Tool

### Features

- Open source, GPL license.
- Written in python (and a bit of C++), supports Linux, Windows, macOS/OSX.
- Support of most VASP file formats.
- Displays structure, unitcell, supercell, charge/probability/spin density, local potential, forces, velocities, dynamics, measure distances and angles.
- Simple structure editor, supports both Cartesian and direct coordinates. Building of supercells and crystal surfaces via cell replication.
- Analyze electronic properties, total and local DOS and band structure.
- Simulation of STM images (constant current and constant height).
- Brillouin zone and K-points visualization.
- Inspection of convergence - size of free energy and selected forces along relaxation path.
- Phonon properties - dispersion, velocity autocorrelation.
- Interactive plots and 3D displays with export to various formats.
- Database support.

<https://github.com/orest-d/p4vasp>

<http://www.p4vasp.at>



-> NSC module

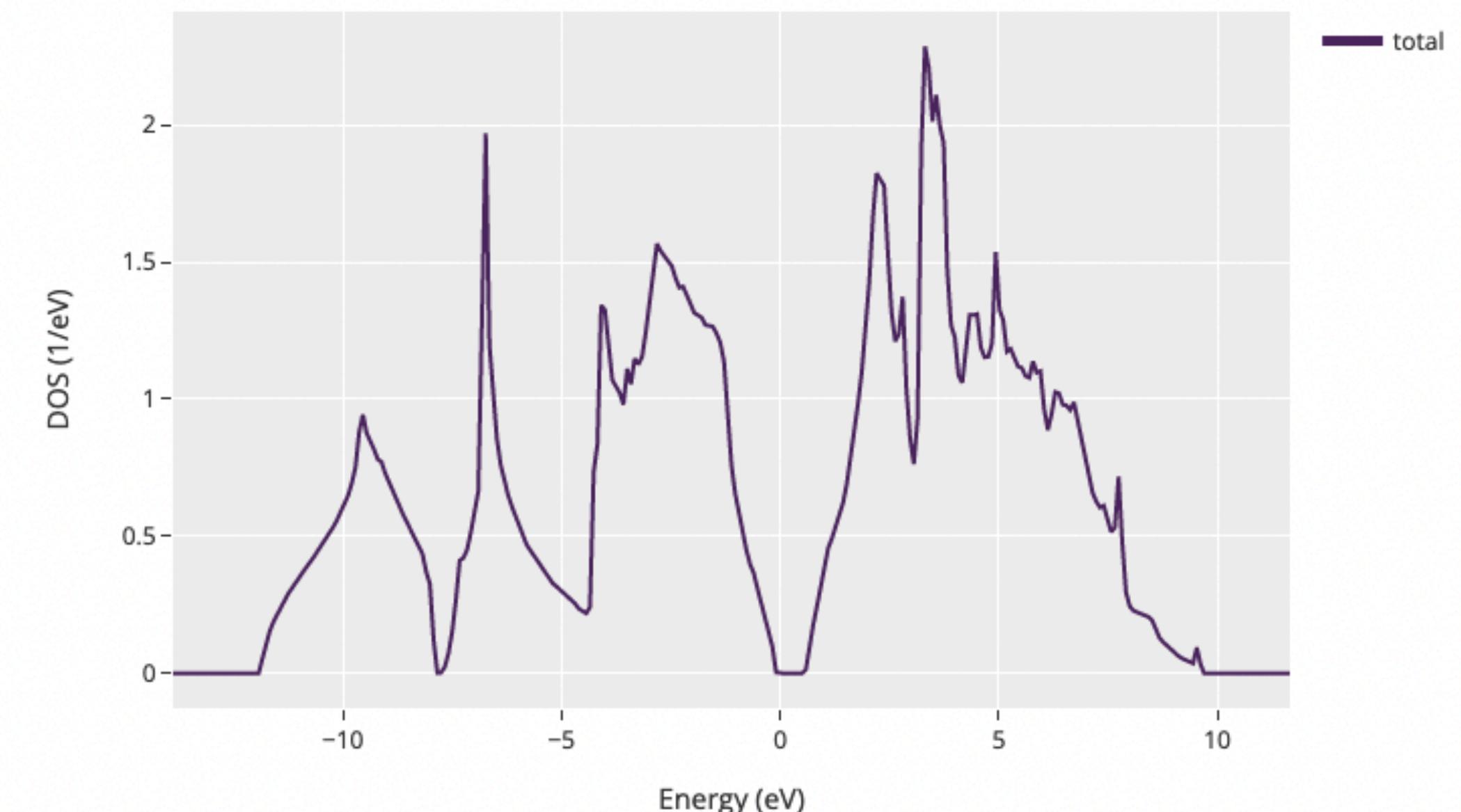
# py4vasp

- Python interface to VASP output
- Quick look at data
- Basis for other postprocessing tools
- VASP  $\geq 6.2$
- Compile with HDF5 (`vaspout.h5`)
- Optimized for use with Jupyter (lab or notebook)



In [1]:

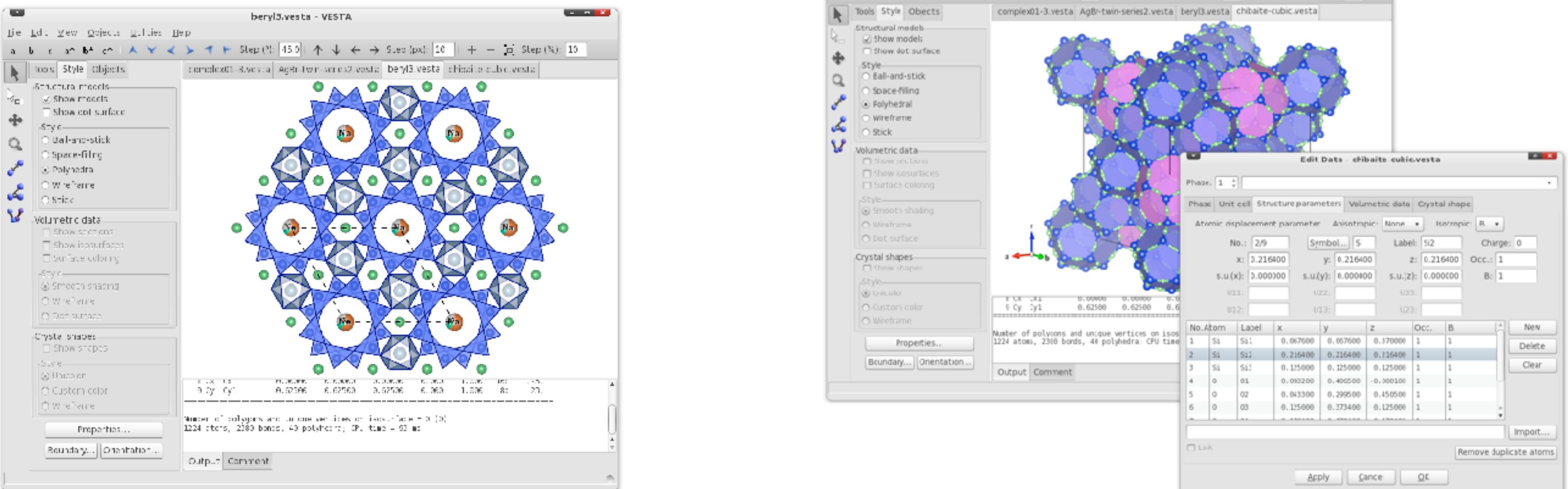
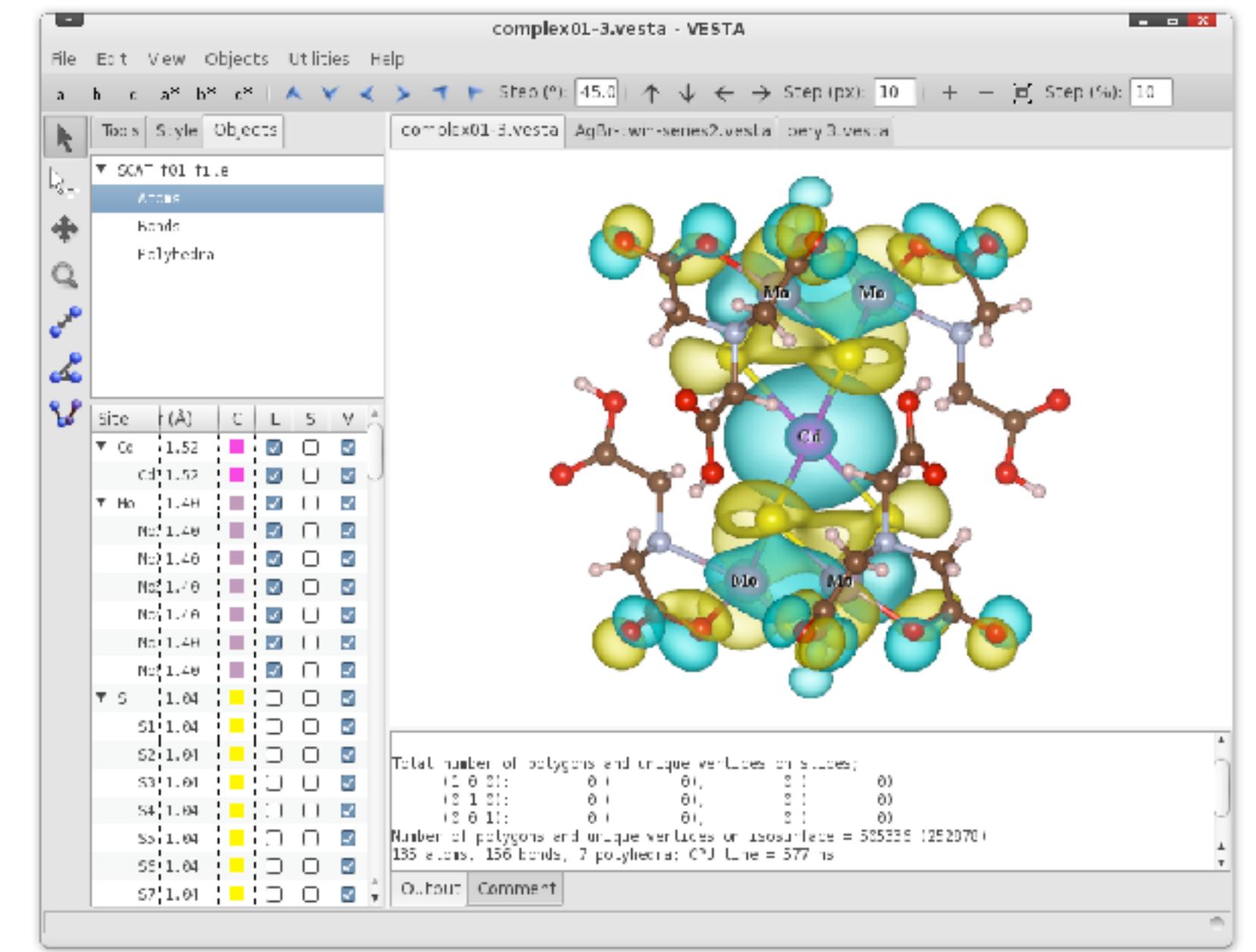
```
import py4vasp
mycalc = py4vasp.Calculation.from_path( "./e04_cd-Si" )
mycalc.dos.plot()
```



<https://www.vasp.at/tutorials/latest/>  
<https://www.vasp.at/py4vasp/latest/>

# VESTA

- Nice/fast tool for visualization
- Lots of functionality
- Export to POSCAR
- Possible to build (check for guides)
- Win/Linux/Mac

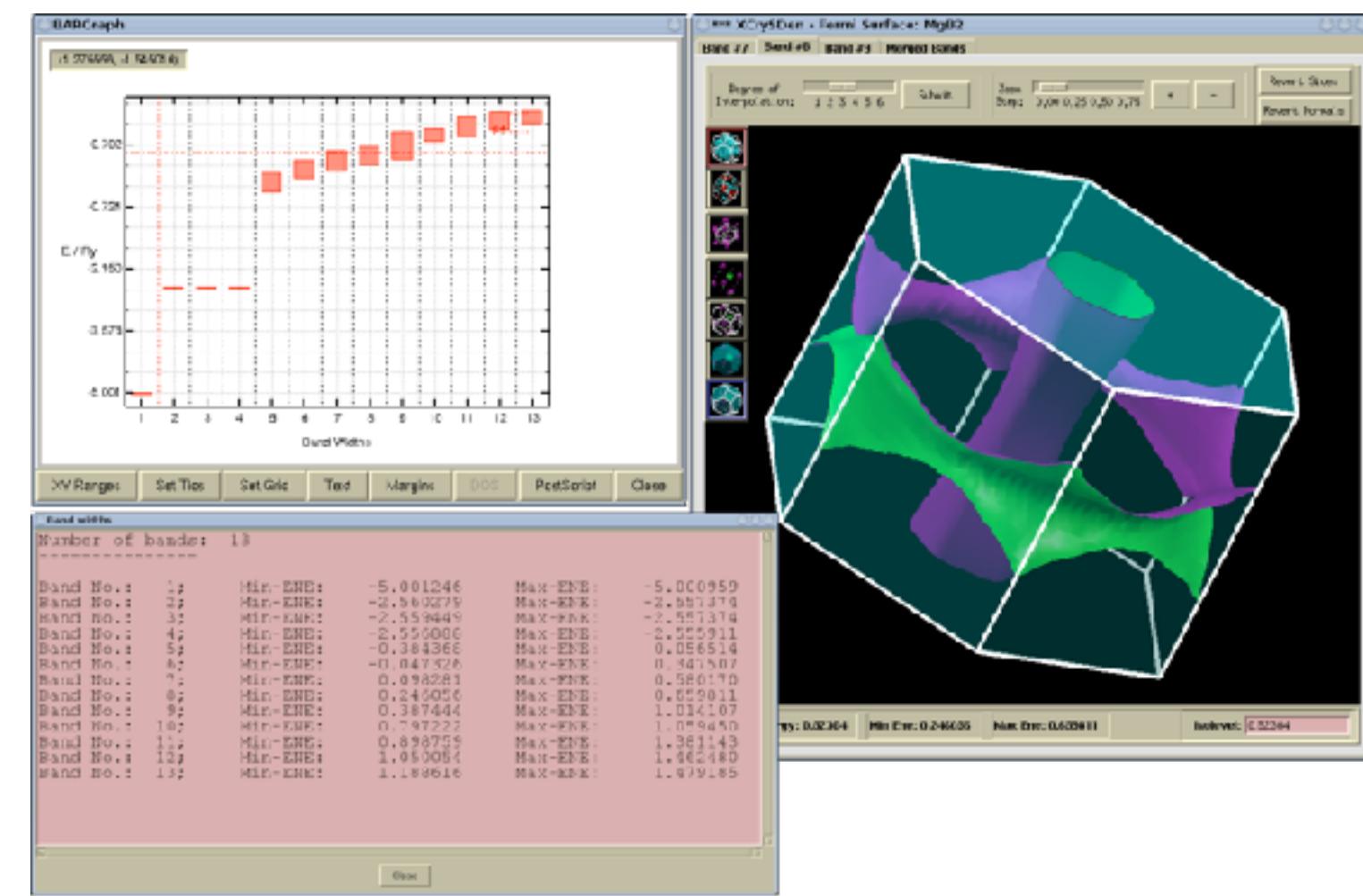
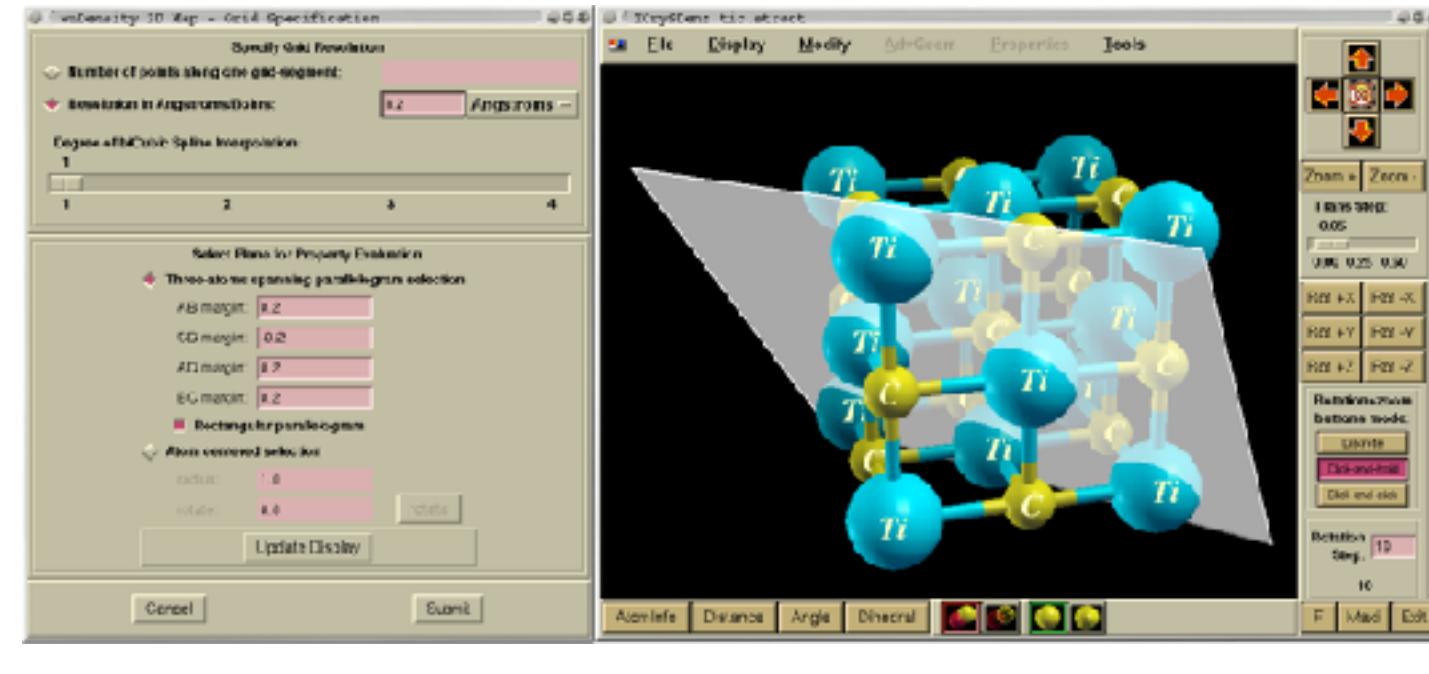


<http://www.jp-minerals.org/vesta/en/>

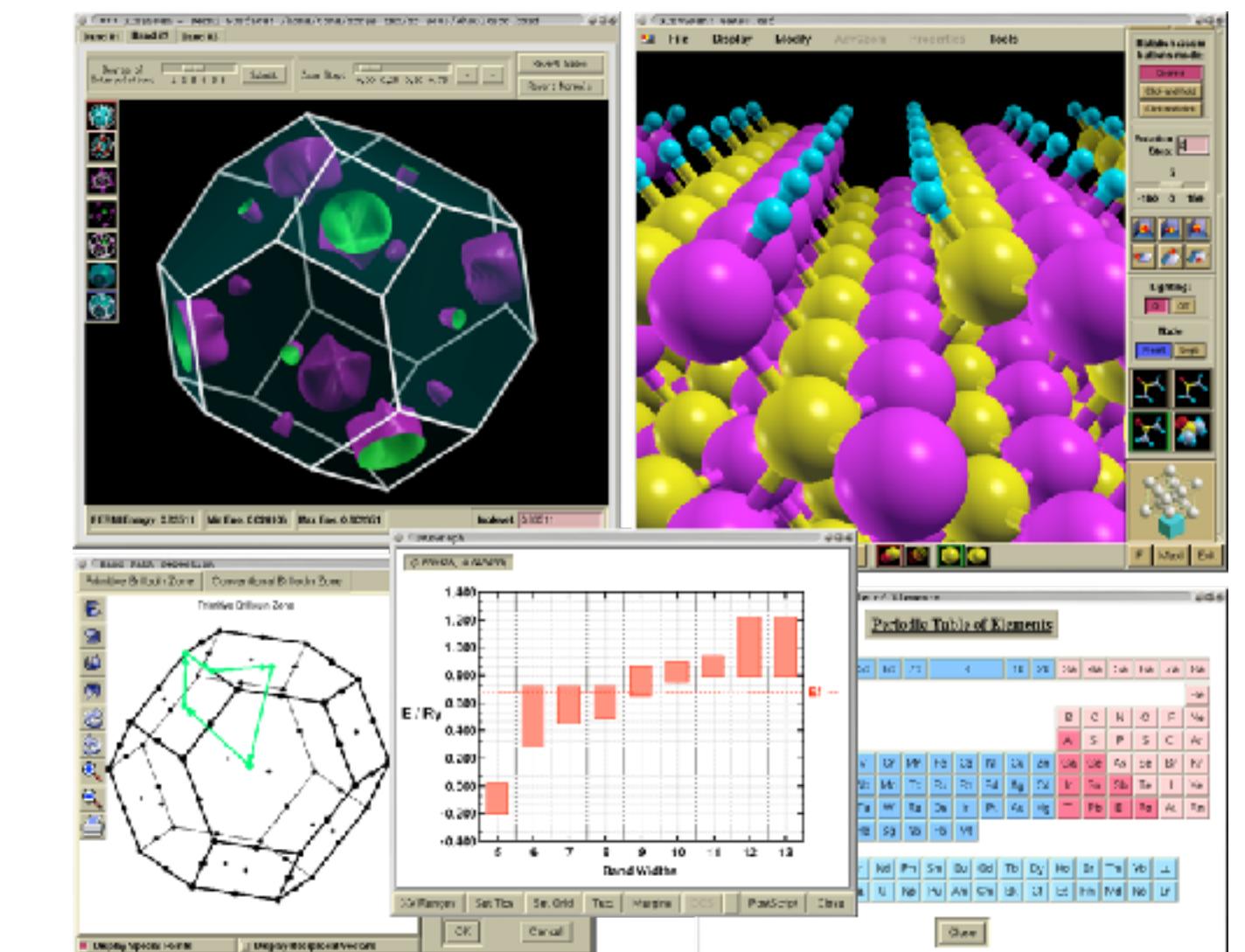
-> NSC module

# XcrysDen

- Crystalline & molecular structure visualization
- Tool for VASP -> XSF



**XCrySDen ...**  
X-window CRYstalline Structures and DENsities



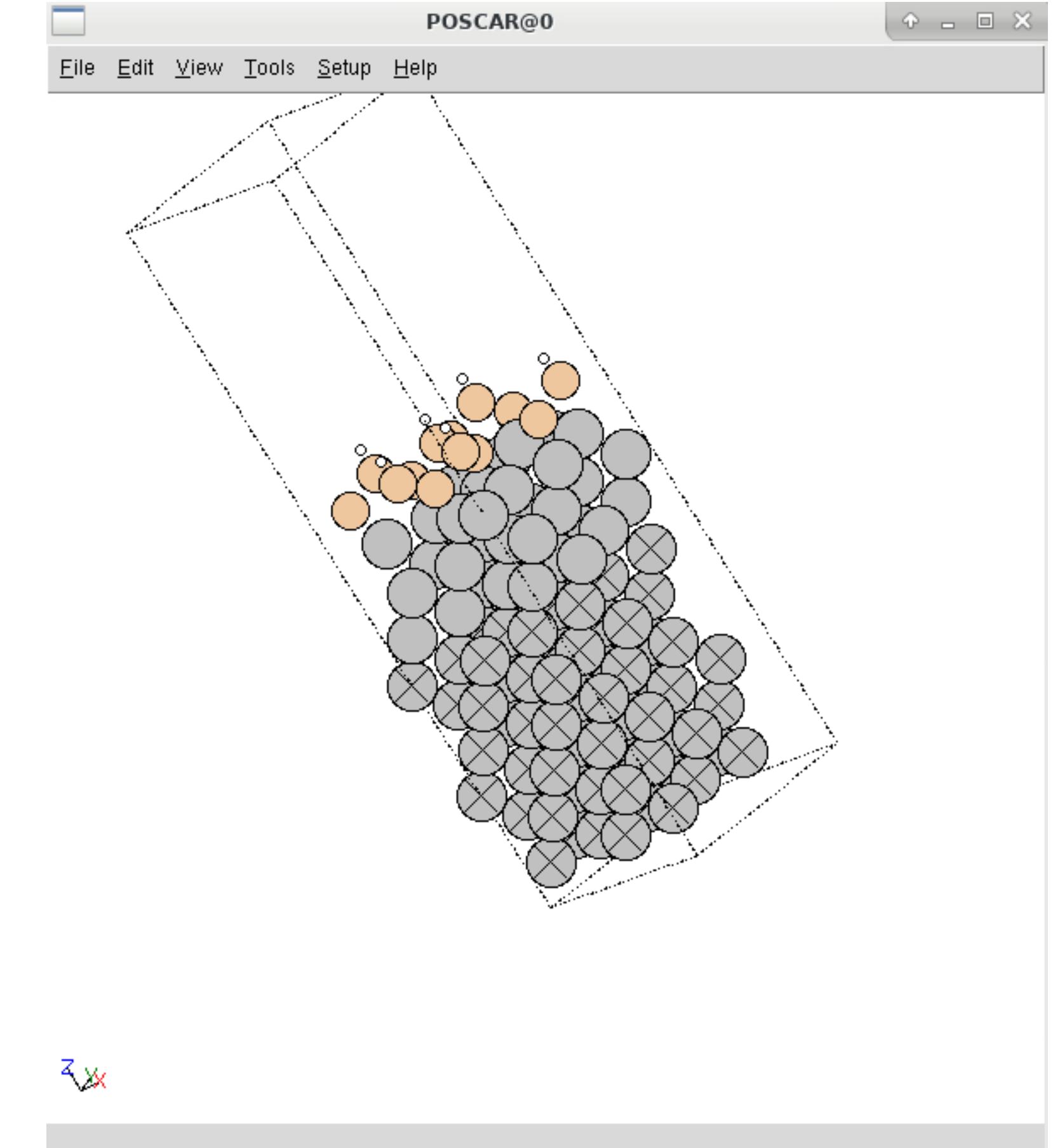
Some examples from <http://www.xcrysden.org/Screenshots.html>

# ASE

- Atomic Simulation Environment (ASE)
- Set of tools & Python modules
- Lots of different functions
- Works with many calculators (e.g. VASP)
- Quickly build supercells, surfaces
  - Also GUI builder, open with: `ase gui`

Al(111) 2x2x3 surface slab with 10Å vacuum  
on each side -> POSCAR

```
from ase.build import fcc111
from ase.io import write
slab = fcc111('Al', size=(2,2,3), vacuum=10.0)
write('POSCAR', slab)
```



# Vasikit

- Analysis tools for VASP
- Structure editor
- DOS, band-structure
- Fermi-surface
- Spin-texture
- Optical properties
- + much more

<https://vaskit.com/>

```
> vaskit
      \\\\
      / _ _ \
      ( | (o)(o) | )   Hey, you must know what you are doing.
      o-----.000o--(())--o000.-----o   Otherwise you might get wrong results.
      |           VASPKIT Standard Edition 1.5.0 (01 Jan. 2024)
      |           Lead Developer: Vei WANG (wangvei@icloud.com)
      |           Main Contributors: Gang TANG, Nan XU & Jin-Cheng LIU
      |           Online Tutorials Available on Website: https://vaskit.com
      o-----.000o-----o
      ( )  Oooo.          VASPKIT Made Simple
      \ ( ( )
      \_)  ) /
      (_)

===== Structural Utilities =====
01) VASP Input-Files Generator    02) Mechanical Properties
03) K-Path for Band-Structure     04) Structure Editor
05) Catalysis-ElectroChem Kit     06) Symmetry Analysis
07) Materials Databases          08) Advanced Structure Models
===== Electronic Utilities =====
11) Density-of-States            21) Band-Structure
23) 3D Band-Structure            25) Hybrid-DFT Band-Structure
26) Fermi-Surface                28) Band-Structure Unfolding
31) Charge-Density Analysis     42) Potential Analysis
44) Piezoelectric Properties     51) Wave-Function Analysis
62) Magnetic Analysis            65) Spin-Texture
68) Transport Properties
===== Misc Utilities =====
71) Optical Properties           72) Molecular-Dynamics Kit
74) User Interface               78) VASP2other Interface
84) ABACUS Interface             91) Semiconductor Kit
92) 2D-Material Kit              95) Phonon Analysis
0) Quit
```

# cif2cell

- Convert .cif to POSCAR
- Many other formats...
- Build supercells quickly
- Lots of functionality, specific VASP options

Check options with:

```
cif2cell -h
```

e.g. download fcc Ag .cif from Crystallography Open Database and test:

```
cif2cell 9008459.cif -p vasp --vasp-format=5 --supercell=[2,2,2]
```

```
cif2cell 9008459.cif -p vasp --vasp-format=5 --supercell=[2,2,2] --no-reduce
```

<https://sourceforge.net/projects/cif2cell/>

<https://pypi.org/project/cif2cell/>

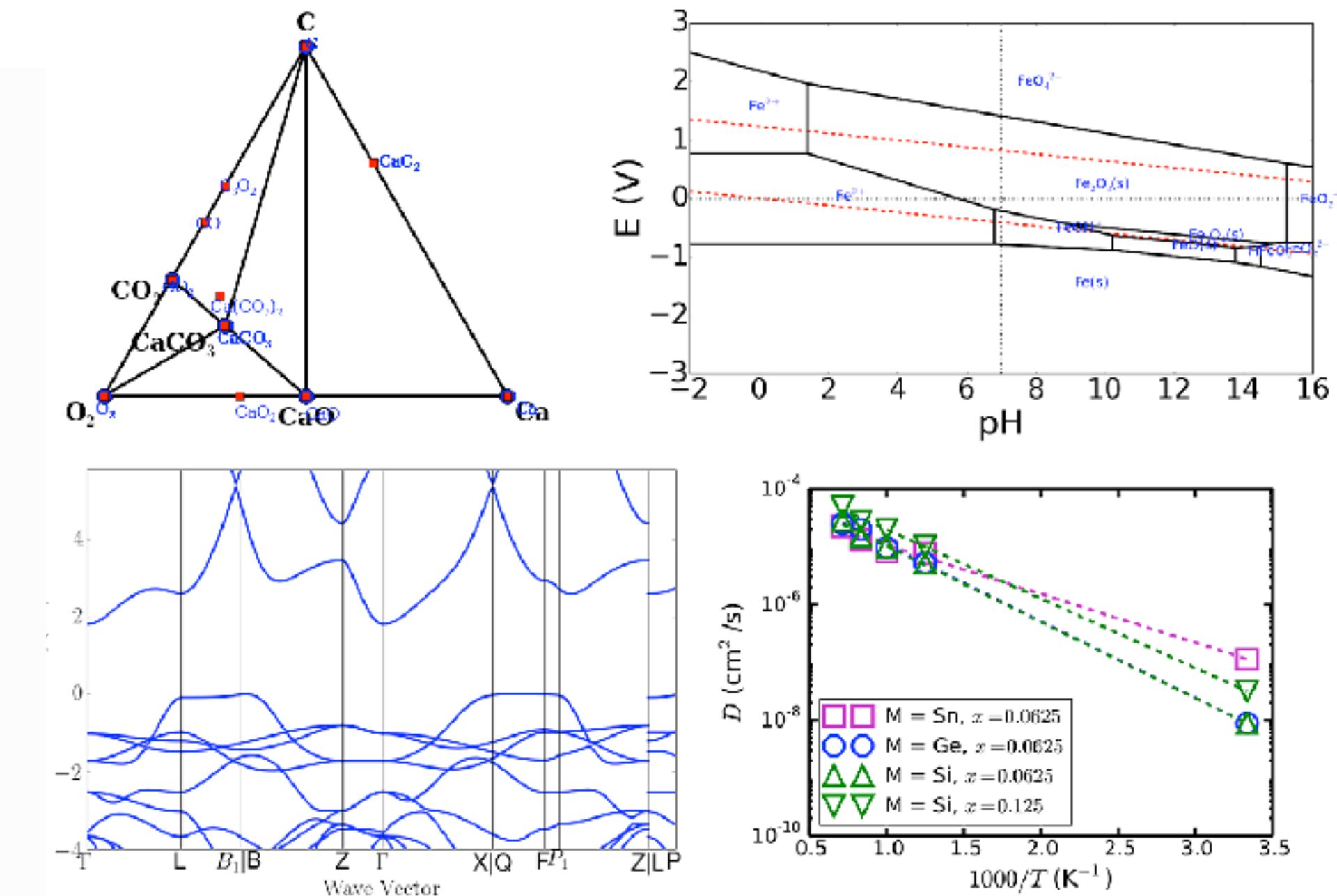
-> NSC module

# pymatgen

## pymatgen

Pymatgen (Python Materials Genomics) is a robust, open-source Python library for materials analysis. These are some of the main features:

1. Highly flexible classes for the representation of Element, Site, Molecule, Structure objects.
2. Extensive input/output support, including support for VASP (<http://cms.mpi.univie.ac.at/vasp/>), ABINIT (<http://www.abinit.org/>), CIF, Gaussian, XYZ, and many other file formats.
3. Powerful analysis tools, including generation of phase diagrams, Pourbaix diagrams, diffusion analyses, reactions, etc.
4. Electronic structure analyses, such as density of states and band structure.
5. Integration with the Materials Project REST API, Crystallography Open Database and other external data sources.



<https://pymatgen.org/>

- > install virtualenv/conda
- > NSC module

# Sumo

## 1. An extensive framework for generating high-symmetry k-point paths.

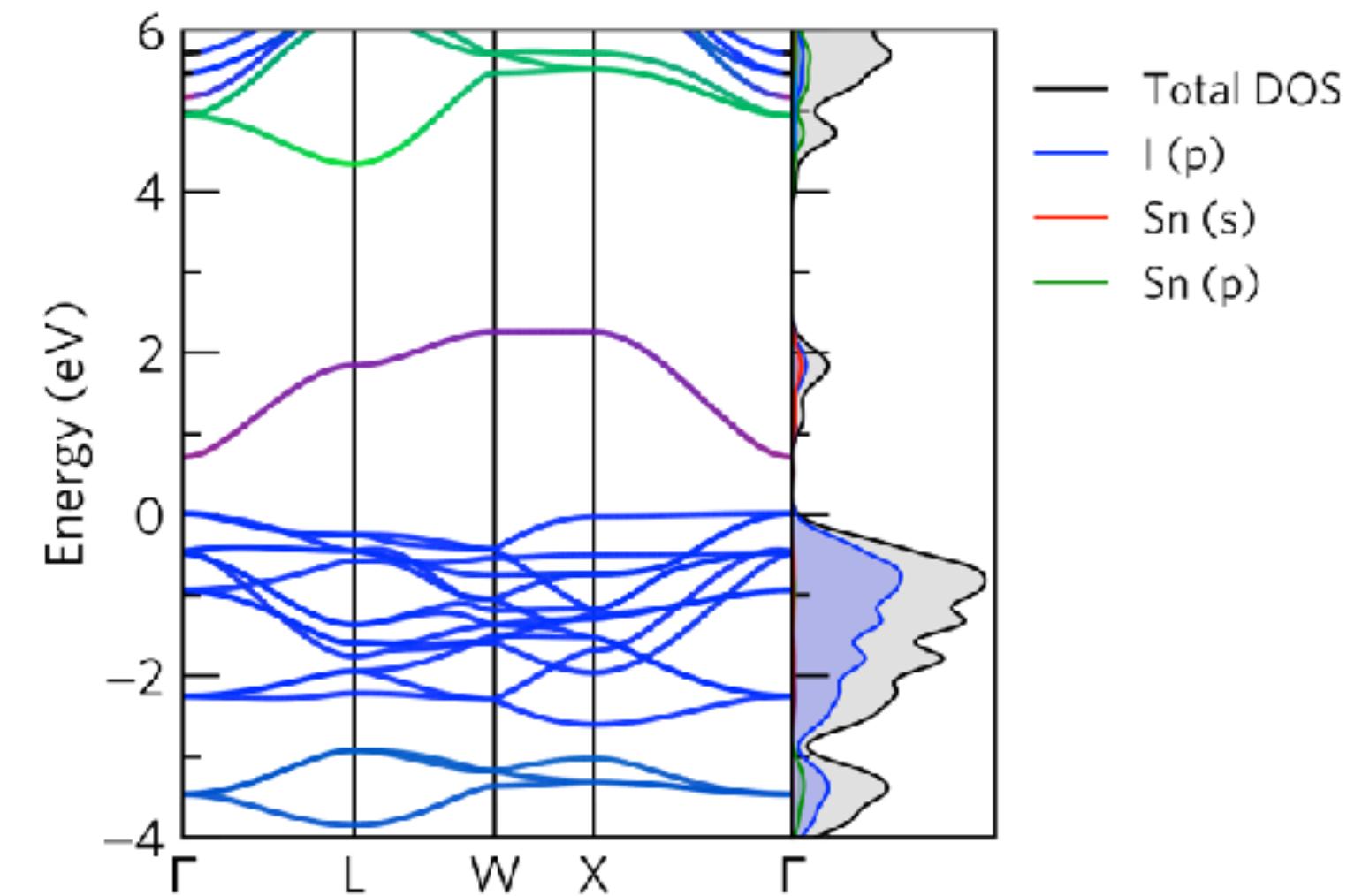
- Crystallographic spacegroups are determined using [Spplib](#).
- Conventional crystallographic paths are built in as well as interfaces to the [SeeK-path](#) and [Pymatgen](#) implementations.

## 2. Plotting scripts for electronic and phonon band structures, density of states, and optical absorption diagrams.

- VASP calculations are imported using [Pymatgen](#).
- The [Phonopy](#) framework is supported for phonon band structures.

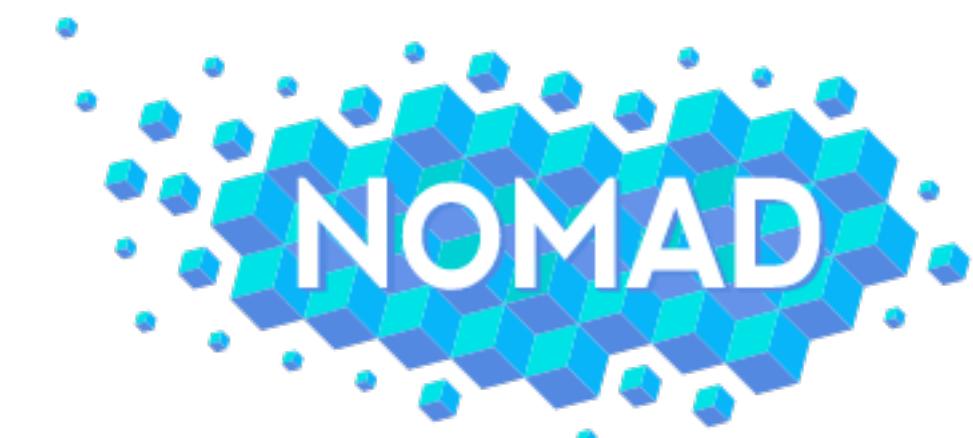
## 3. Analysis scripts to calculate parabolic and non-parabolic band effective masses.

- Curve fitting is performed using [Scipy](#).



# More tools & resources...

- SeeK-path: k-path finder and visualizer, also on Materials cloud
- Grace (xmGrace): quick plotting tool
- NOMAD: computational materials science data
- Atomsk: create & handle structures
- c2x: e.g. plot Fermi surface
- Occupation matrix control in VASP → NSC module
- ICSD crystallographic database
- Bilbao crystallographic server
- Crystallography Open Database



...probably missed many other interesting resources, so look around!

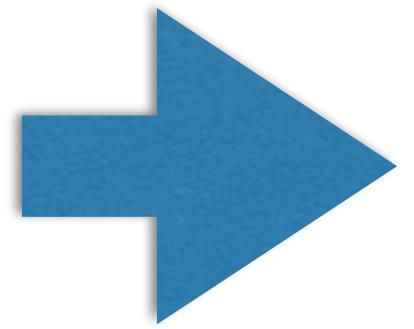
# Summary

A repetition & some final points

# Summary “rules of thumb”

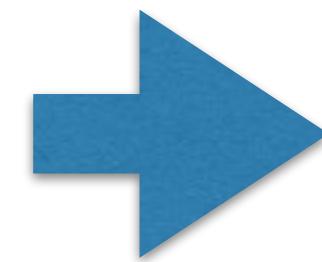
- job size (max): **total cores  $\approx$  NBANDS / 8**
- **NSIM = 4** (default), or more (2 Beskow)
- **NCORE = cores/node**
- **PREC = Accurate** - if forces important
- **ENCUT = ENMAX x1.5** - “max setting”
- **KPAR = min (nodes, k-points)** HSE06, especially useful
- In general, INCAR default settings OK
- GPU: important to increase NSIM

# New VASP users

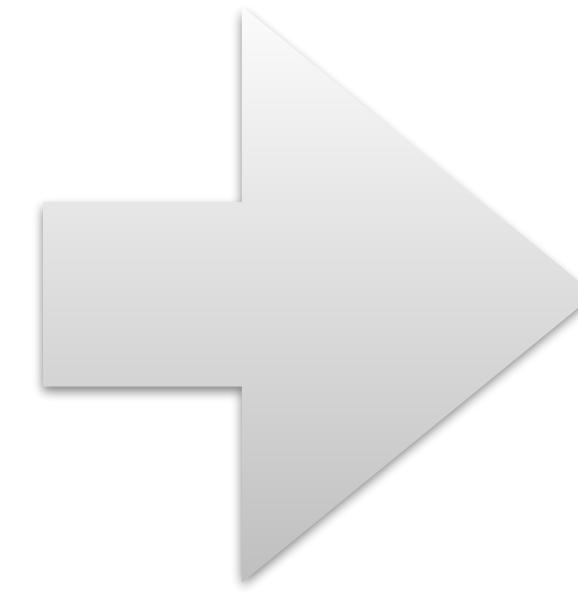


- “Play around” - test different options
- Check the material on the VASP wiki
- What is the “size” of the problem?
- What level of accuracy is needed?
- Can VASP solve problem X?

# Resources



- Wiki and Manual  
**Check in detail!**
- Examples, tutorials
- Presentations
- Forum



Find all the links:  
<https://vasp.at/>

- Also other resources, materials and tools for VASP (see this presentation)
- Peter Larsson's old blog at NSC: <https://www.nsc.liu.se/~pla/>
- NSC VASP installations: <https://www.nsc.liu.se/software/installed/tetralith/vasp/>

Questions / trouble @NSC clusters? [support@nsc.liu.se](mailto:support@nsc.liu.se)