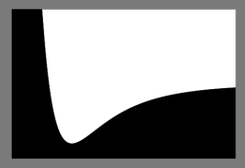


# *The KIM Calculator in ASE:*
## *Reproducible Access to Reliable Classical Interatomic Models*


Jakob Schiøtz[1], Daniel S. Karls[2], Mingjian Wen[3], Ryan S. Elliott[2], Ellad B. Tadmor[2]

[1] Department of Physics, Technical University of Denmark

[2] Department of Aerospace Engineering and Mechanics, University of Minnesota

[3] Energy Technologies Area, Lawrence Berkeley National Laboratory



NSF CDI (2009–2014); NSF CDS&E (2014–2018); NSF CMMT (2019–)

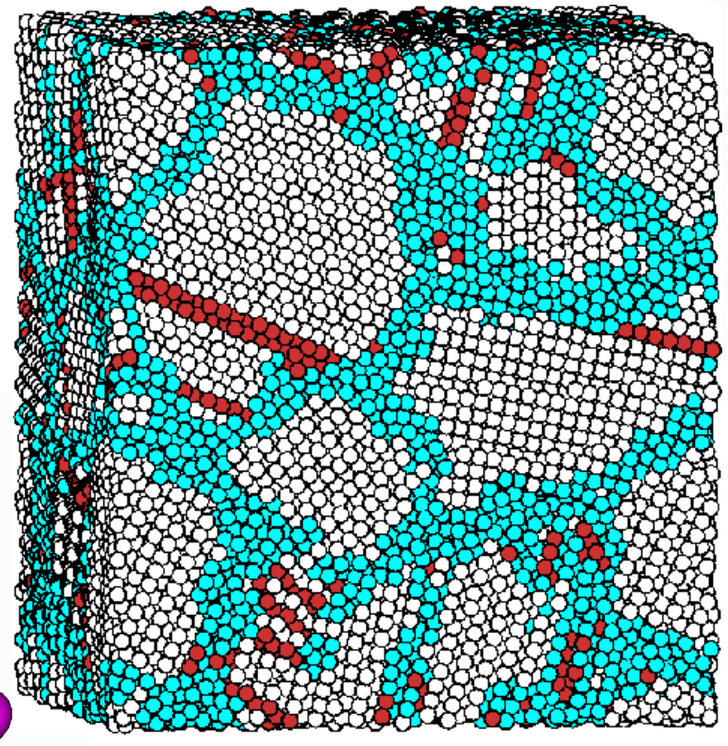

---

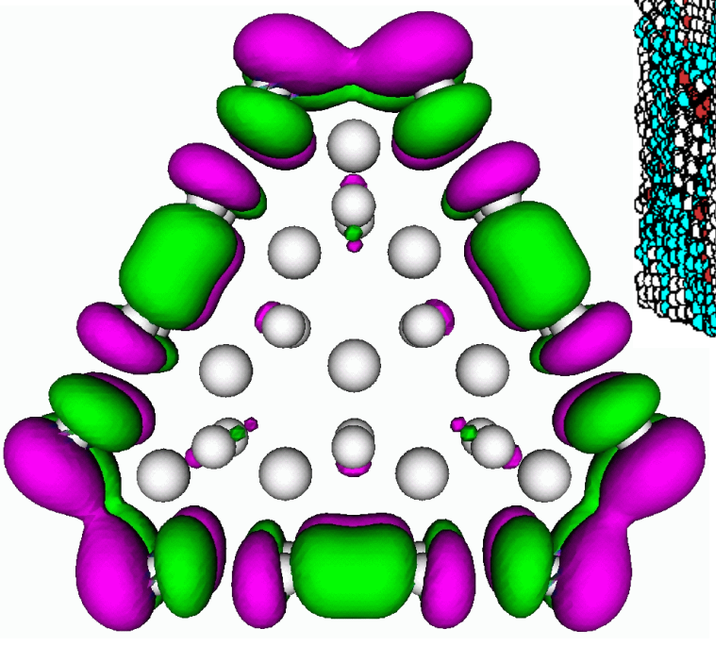

ASE Workshop, Gothenburg, Sweden, November 19–22, 2019

# Simulations of Materials - length scales

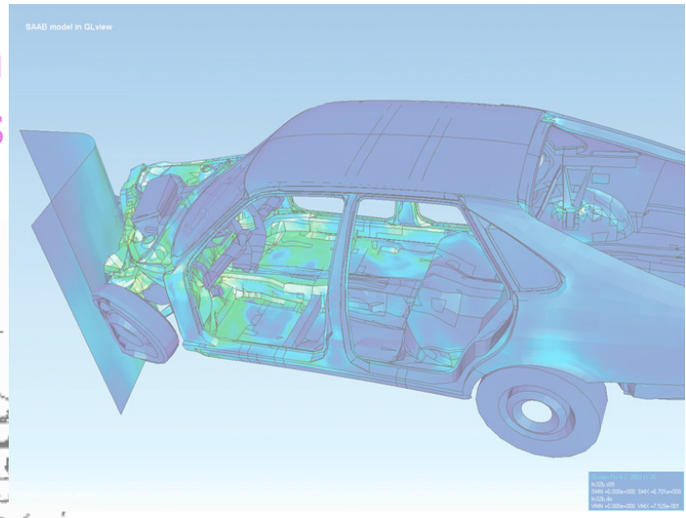
**This talk**

**ASE**





Continuum Dynamics




Defect dynamics: Solve equation of motion of the defects (dislocations etc).

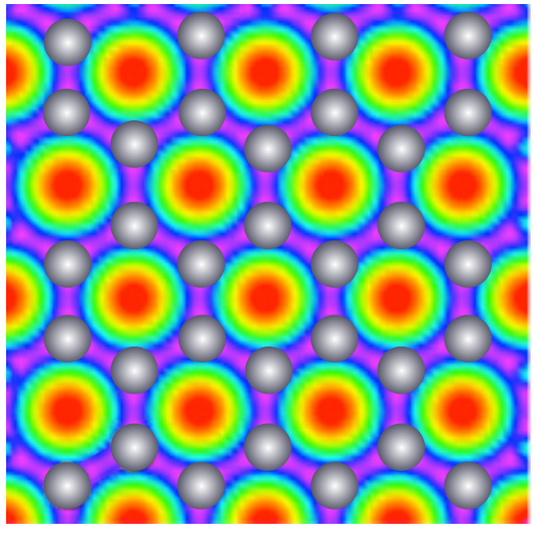
Classical Monte Carlo and Molecular Dynamics: Describe atoms as classical particles. Millions of atoms.

Quantum mechanics: solve Schrödinger's equation for the electrons. Very precise, very slow. Hundreds of atoms.

# Classical Interatomic Models (IM)

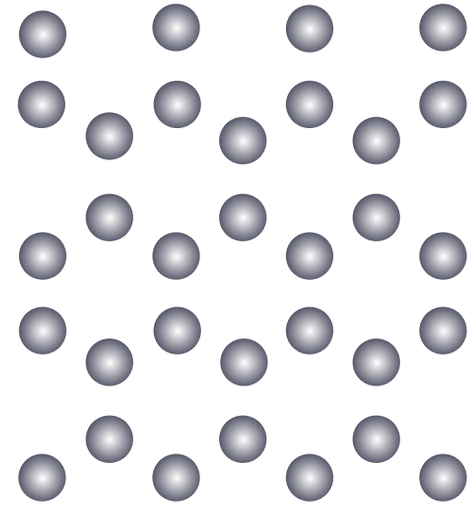
**ASE** is often seen as an interface to DFT codes, however support is growing for classical interatomic models (IMs).

quantum view of bonding



Integrate out the electrons

classical interatomic model



Training Set

DFT results

Experimental reference data

best fit

$$\widetilde{\mathcal{V}}(\boldsymbol{r}^1, \boldsymbol{r}^2, \ldots, \boldsymbol{r}^N; \alpha_1, \alpha_2, \ldots)$$

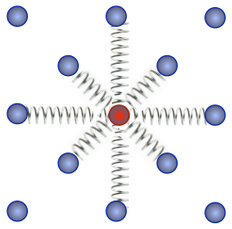
positions of nuclei

fitting parameters

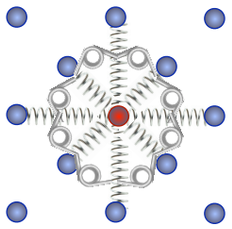
# Menagerie of Interatomic Models (IMs)

- Many different kinds of IMs have been developed over the years:

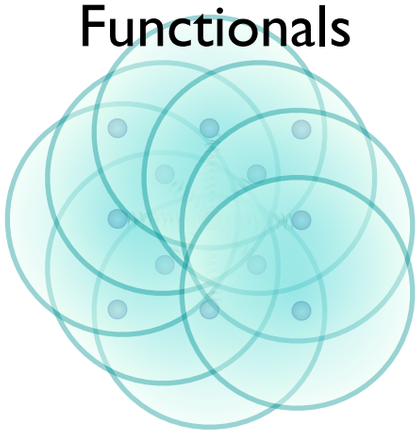
Pair Potentials



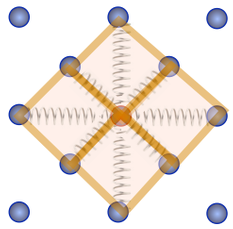
Cluster Potentials



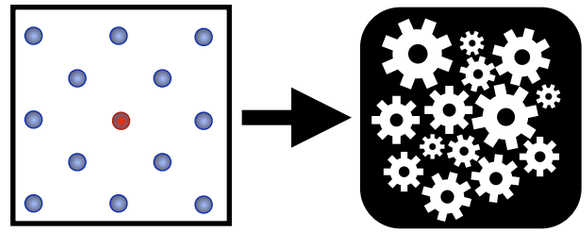
Pair/Cluster Functionals



Bond-order Potentials



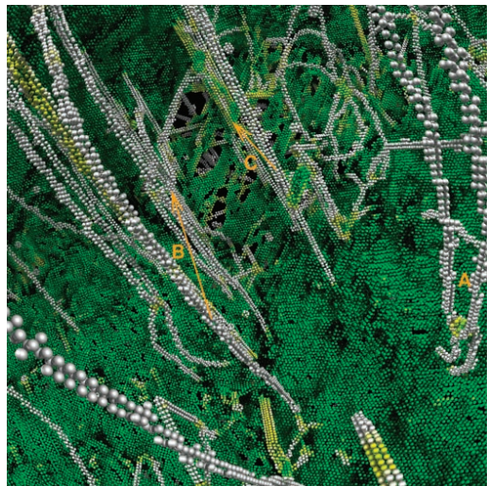
Machine learning Potentials



- The reduced cost of IMs makes it possible to simulate much larger systems and study phenomena inaccessible to DFT.

  *Dislocations in a grain of nanocrystal copper. Molecular Dynamics simulation with 100 million atoms.*

  Schiøtz and Jacobsen, *Science*, 301:1357, 2003.




- However the proliferation of IMs and classical simulation codes have created problems that the **Knowledgebase of Interatomic Models (KIM)** project is attempting to address.

# Workflow: Simulating a new material

**A reactive potential for hydrocarbons with intermolecular interactions**

Steven J. Stuart[a)]
*Department of Chemistry, Clemson University, Clemson, South Carolina 29634*
Alan B. Tutein[b)] and Judith A. Harrison[c)]
*Department of Chemistry, U.S. Naval Academy, Annapolis, Maryland 21402*

1. Identify one (or more) interatomic models in the literature.
   - Select the one you think is most promising.
2. Code the equations in your favorite MD code (ASE, of course!).
   - Also code the derivatives (the forces)

$$E^{\text{REBO}}_{ij} = V^R_{ij}(r_{ij}) + b_{ij} V^A_{ij}(r_{ij}),$$

$$V^{\text{LJ}}_{ij}(r_{ij}) = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right].$$

$$E^{\text{LJ}}_{ij} = S(t_r(r_{ij}))S(t_b(b^*_{ij}))C_{ij}V^{\text{LJ}}_{ij}(r_{ij}) + [1 - S(t_r(r_{ij}))]C_{ij}V^{\text{LJ}}_{ij}(r_{ij}),$$

where $S(t)$ is a universal switching function,

$$S(t) = \Theta(-t) + \Theta(t)\Theta(1-t)[1 - t^2(3-2t)],$$

$$t_r(r_{ij}) = \frac{r_{ij} - r^{\text{LJ min}}_{ij}}{r^{\text{LJ max}}_{ij} - r^{\text{LJ min}}_{ij}},$$

$$b^*_{ij} = b_{ij}|_{r_{ij}=r^{\min}_{ij}}.$$

$$w_{ij}(r_{ij}) = S'(t_c(r_{ij}))$$

with

$$t_c(r_{ij}) = \frac{r_{ij} - r^{\min}_{ij}}{r^{\max}_{ij} - r^{\min}_{ij}}.$$

$$E = E^{\text{REBO}} + E^{\text{LJ}} + E^{\text{tors}},$$

$$t_b(b_{ij}) = \frac{b - b^{\min}_{ij}}{b^{\max}_{ij} - b^{\min}_{ij}},$$

$$E^{\text{tors}} = \frac{1}{2}\sum_i \sum_{j\neq i} \sum_{k\neq i,j} \sum_{l\neq i,j,k} w_{ij}(r_{ij})w_{jk}(r_{jk})w_{kl}(r_{kl}) \times V^{\text{tors}}(\omega_{ijkl}).$$

$$C_{ij} = 1 - \max\{w_{ij}(r_{ij}), w_{ik}(r_{ik})w_{kj}(r_{kj}), \forall k \; w_{ik}(r_{ik})w_{kl}(r_{kl})w_{lj}(r_{lj}), \forall k,l\}.$$

$$S'(t) = \Theta(-t) + \Theta(t)\Theta(1-t)\frac{1}{2}[1 + \cos(\pi t)].$$

$$V = \frac{1}{2}\sum_{k=1}^{3} V_k[1 - (-1)^k \cos(k\omega)].$$

$$T(\omega) = 3V(\omega) + 3V\left(\omega + \frac{2\pi}{3}\right) + 3V\left(\omega - \frac{2\pi}{3}\right) = \frac{1}{2}\epsilon[1 + \cos(3\omega)].$$

$$V^{\text{tors}}(\omega) = \epsilon\left[\frac{256}{405}\cos^{10}\left(\frac{\omega}{2}\right) - \frac{1}{10}\right].$$

# Workflow: Simulating a new material

1. Identify one (or more) interatomic models in the literature.
   - Select the one you think is most promising.
2. Code the equations in your favorite MD code (ASE, of course!).
   - Also code the derivatives (the forces)
3. Test it.
4. Debug it.
5. If no parameters: Fit to DFT and experimental data.
6. Evaluate the model on known, relevant materials properties.
   - If it is good: **Do some science**.
   - If it is bad: **Go back to step 1**.

# OpenKIM Workflow: Simulating a new material

1. Identify one (or more) interatomic models on OpenKIM.org
   - Select the one that reproduce relevant materials parameters the best.
2. Download and compile the model
   - Your favorite MD program can directly use it (ASE, ASAP, LAMMPS)
3. (Test it).
4. ~~Debug it.~~
5. If no parameters: Fit to DFT and experimental data – upload it.
6. Evaluate the model on known, relevant materials properties.
   - If it is good: **Do some science**.
   - If it is bad: **Go back to step 1**.

# Why KIM?

- The KIM effort addresses key problems faced by molecular modelers:

**Problem 1:** It is currently very difficult or even impossible for a researcher to reproduce published results obtained using molecular simulations with a given IM or to use the same IM in a different study.

Archival Storage  DOIs

**Problem 2:** It is very difficult to port IMs between simulation codes that have different program architectures and may be written in different computer languages.

KIM API

**Problem 3:** A great deal of researcher time is spent redeveloping methods for computing complex material properties (e.g. melting temperature, thermal conductivity, phonon spectra, entropic properties, phase diagrams, etc.).

KIM Tests

# Why KIM?

- The KIM effort addresses key problems faced by molecular modelers:

**Problem 4:** Programming errors in IM implementations can lead to systematic errors that can go unnoticed, or result in strange behavior that can be extremely difficult and time-consuming to debug.

KIM Verification Checks

**Problem 5:** Researchers do not have a central location to exchange information about specific IMs and share analysis and visualization tools that they have developed related to molecular simulation.

KIM Model Pages  Central Discussion Forum

**Problem 6:** Researchers do not have easy and reliable access to predictions of an IM for certain properties needed for problem setup or analysis, e.g. an equilibrium lattice constant is required to build a crystal or a bulk cohesive energy is required in a surface energy calculation.

KIM Queries

# Knowledgebase of Interatomic Models

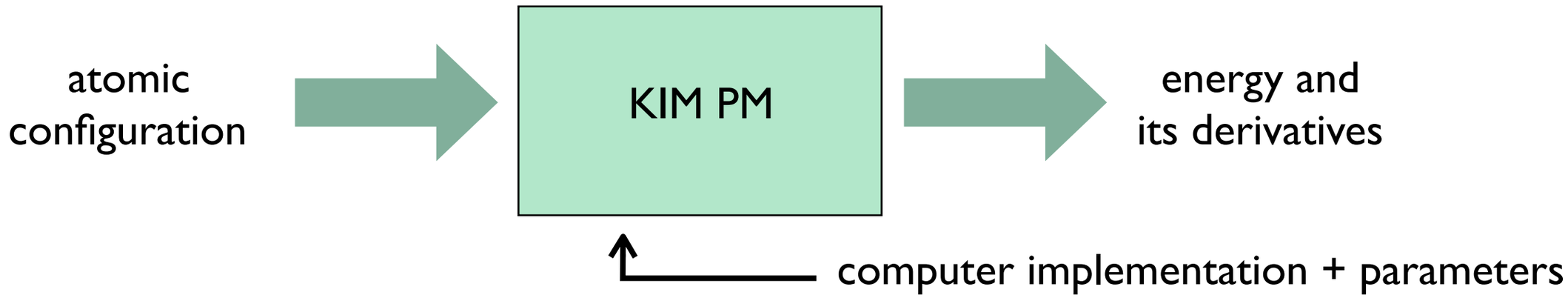
The Open Knowledgebase of Interatomic Models (KIM) is a cyberinfrastructure funded by the U.S. National Science Foundation (NSF) with the following features:

- Curated repository of interatomic models (IMs) (potentials and force fields) with comprehensive provenance and version control.
- Application Programming Interface (API) standards connecting molecular simulation codes ("simulators") with IMs.
- Standardized testing framework for archived IMs including their predictions for material properties and checks on their coding integrity.
- Source and binary distribution framework for easy installation and use of the KIM API and KIM IMs with conforming simulators.
- Rigorous transferability and uncertainty estimation for KIM IMs based on machine learning approaches to select IMs for target application and providing error bounds on their predictions (under development).

**Funding:** NSF CDI (2009-2014); NSF CDS&E (2014-2018); NSF CMMT (2019-)

# Types of KIM Models

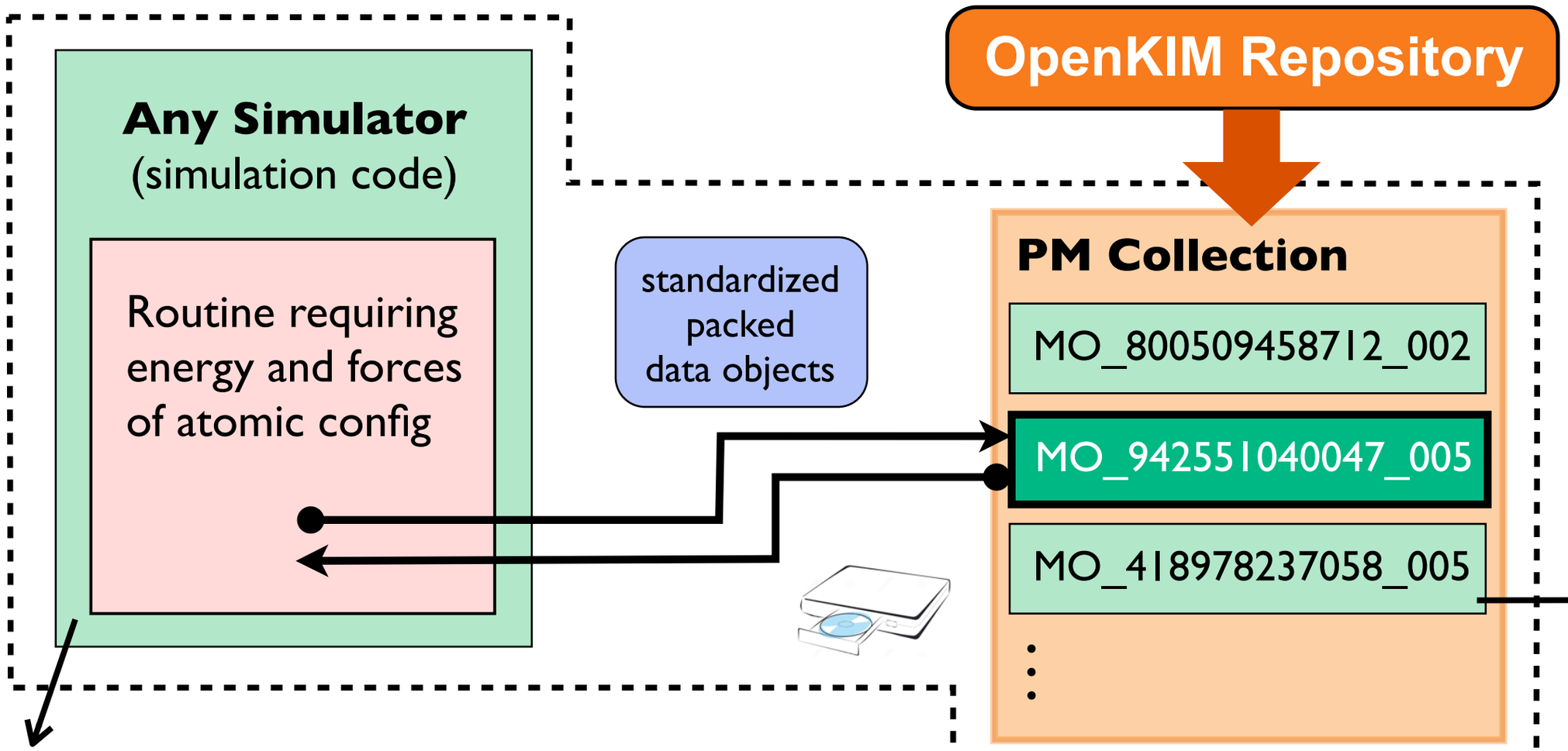
- KIM has two types of Models:
  - KIM Portable Model (PM): An autonomous interatomic model that works seamlessly with any simulation code that conforms to KIM standards.
    - A PM is independent computer implementation of an interatomic model written in one of the languages supported by KIM (C, C++, Fortran).




  - KIM Simulator Model (SM): An interatomic model that only works with a single simulation code (simulator) in which it is implemented.
    - An SM is a package curated on OpenKIM that includes all the necessary parameter files, simulator input commands, and metadata (supported species, units, etc.) needed to run model in its native simulator.

# KIM API — Portable Models

- KIM Models employ the KIM Application Programming Interface (API).
  - Portable Models (PMs) conform to the KIM API Portable Model Interface (PMI)

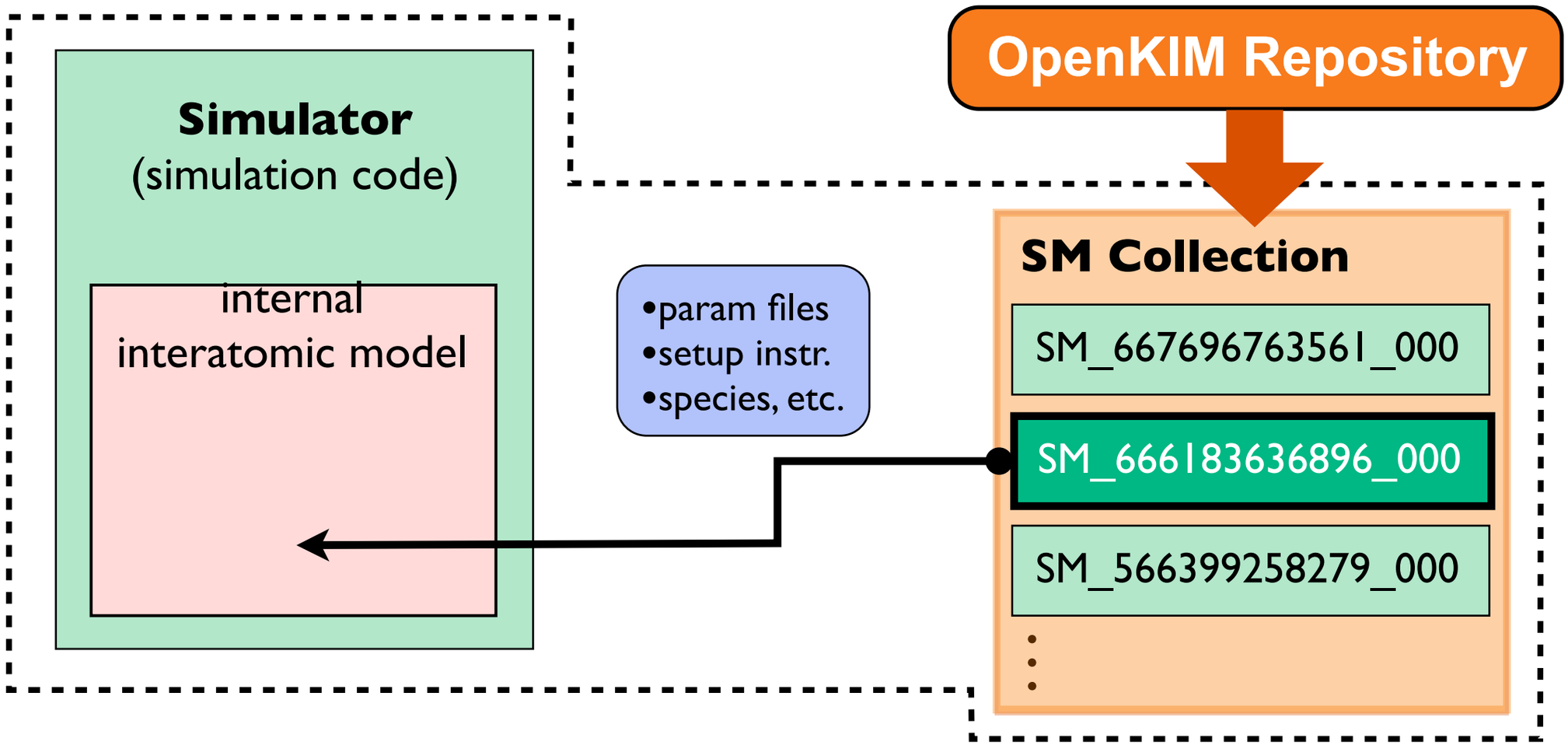



- Stand-alone simulation computer program (MD, MC, lattice dynamics, multiscale, etc.)
- Can be written in any language supported by the API (Fortran, C, C++, Python)

- Autonomous code that given a set of atomic positions, species, ... computes energy, forces, ...
- Can be written in any language supported by the API (Fortran, C, C++)

# KIM API — Simulator Models

- KIM Models employ the KIM Application Programming Interface (API).
  - Simulator Models (SMs) conform to the KIM API Simulator Model Interface (SMI)




- The KIM API v2 is designed with simplicity in mind and adheres to API best practices:

| | | |
|---|---|---|
| • Implementation hiding (pimpl idiom) | • Ease of use (discoverable, consistent, orthogonal) | • Use of namespaces |
| • Loose coupling | • Static factory methods | • Const-correctness |
| • Minimal-completness | | • Avoid abbreviations |

# KIM Testing Framework

- All KIM IMs are subjected to Verification Checks (VCs) for coding integrity

| Mandatory | Consistency | Informational |
|---|---|---|
| ▪ Species supported as stated;<br>▪ Unit conversion handled correctly;<br>▪ Domain decomposition handled correctly;<br>▪ ... | ▪ Numerical derivative check of forces, virial, hessian, ...;<br>• Translational and rotational invariance;<br>▪ ... | ▪ Continuity, smooth cutoff;<br>• Inversion symmetry;<br>▪ Coding issues: memory leaks, optimization dependence, ...<br>▪ ... |

- All KIM IMs are run against all compatible KIM Tests to compute material properties:

| Bulk | Wall | Line |
|---|---|---|
| - cohesive energy<br>- elastic constants<br>- lattice constants<br>- phonon spectrum<br>- thermal conductivity<br>- thermal expansion<br>- .. | - antiphase boundary<br>- gamma surface<br>- grain boundary structure<br>- stacking fault energy<br>- surface energy<br>- surface structure<br>- ... | - dislocation core structure<br>- dislocation core energy<br>- Peierls barrier<br>- ...<br>**Point**<br>- vacancy formation energy<br>- vacancy migration barrier |

# Models on openkim.org

OpenKIM | Getting Started | About | Download/Upload | Browse | Support | tadmor | 1

## Welcome to the **K**nowledgebase of **I**nteratomic **M**odels!

OpenKIM is an online framework for making molecular simulations reliable, reproducible, and portable. Computer implementations of interatomic models are archived in OpenKIM, verified for coding integrity, and tested by computing their predictions for a variety of material properties. Models conforming to the KIM *application programming interface* (API) work seamlessly with major simulation codes that have adopted the KIM API standard.

> "*All models are wrong but some are useful.*"
> — George E. P. Box

Models | Tests | Contribute a Model or Data

### Models

Click on an element to find interatomic models for that species. You can narrow the selection to models that support multiple species after you click.

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| H | Sp | | | | | | | | | | | | | | | | He |
| Li | Be | | | | | | | | | | | B | C | N | O | F | Ne |
| Na | Mg | | | | | | | | | | | Al | Si | P | S | Cl | Ar |
| K | Ca | Sc | Ti | V | Cr | Mn | Fe | Co | Ni | Cu | Zn | Ga | Ge | As | Se | Br | Kr |
| Rb | Sr | Y | Zr | Nb | Mo | Tc | Ru | Rh | Pd | Ag | Cd | In | Sn | Sb | Te | I | Xe |
| Cs | Ba | | Hf | Ta | W | Re | Os | Ir | Pt | Au | Hg | Tl | Pb | Bi | Po | At | Rn |
| Fr | Ra | | Rf | Db | Sg | Bh | Hs | Mt | Ds | Rg | Cn | Nh | Fl | Mc | Lv | Ts | Og |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| La | Ce | Pr | Nd | Pm | Sm | Eu | Gd | Tb | Dy | Ho | Er | Tm | Yb | Lu |
| Ac | Th | Pa | U | Np | Pu | Am | Cm | Bk | Cf | Es | Fm | Md | No | Lr |

# Models on openkim.org

Cu

| Extended KIM ID | Title |
|---|---|
| EAM_Dynamo_AcklandTichyVitek_1987_Cu__MO_179025990738_005 | Finnis-Sinclair potential (LAMMPS cubic hermite tabulation) for Cu developed by Ackland et al. (1987) v005 |
| EAM_Dynamo_AcklandTichyVitek_1987v2_Cu__MO_762798677854_000 | Finnis-Sinclair potential (LAMMPS cubic hermite tabulation) for Cu developed by Ackland et al. (1987), version 2 refitted for radiation studies v000 |
| EAM_Dynamo_AcklandVitek_1990_Cu__MO_642748370624_000 | Finnis-Sinclair potential (LAMMPS cubic hermite tabulation) for Cu developed by Ackland and Vitek (1990) v000 |
| EAM_Dynamo_AdamsFoiles_1989Universal6_Cu__MO_145873824897_000 | EAM potential (LAMMPS cubic hermite tabulation) for Cu (Universal6) developed by Adams and Foiles (1989) v000 |

- 
- 
- 

| | |
|---|---|
| EAM_Dynamo_ZhouWadleyJohnson_2001_CuTa__MO_547744193826_000 | EAM potential (LAMMPS cubic hermite tabulation) for the Cu-Ta system developed by Zhou, Wadley and Johnson (2001) v000 |
| EAM_Dynamo_ZhouWadleyJohnson_2001_Cu__MO_280823813353_000 | EAM potential (LAMMPS cubic hermite tabulation) for Cu developed by Zhou, Wadley and Johnson (2001) v000 |
| EAM_NN_Johnson_1988_Cu__MO_887933271505_002 | EAM Potential (analytical nearest-neighbor) for Cu developed by Johnson (1988) v002 |
| EMT_Asap_MetalGlass_BaileySchiotzJacobsen_2004_CuMg__MO_228059236215_001 | EMT potential for Cu-Mg metallic glasses developed by Bailey, Schiotz, and Jacobsen (2004) v000 |
| EMT_Asap_MetalGlass_CuMgZr__MO_655725647552_002 | Effective Medium Theory potential for CuMg and CuZr alloys, in particular metallic glasses. |
| EMT_Asap_MetalGlass_PaduraruKenoufiBailey_2007_CuZr__MO_987541074959_001 | EMT potential for Cu-Zr metallic glasses developed by Paduraru et al. (2007) v000 |
| EMT_Asap_Standard_JacobsenStoltzeNorskov_1996_AlAgAuCuNiPdPt__MO_115316750986_001 | EMT potential for Al, Ni, Cu, Pd, Ag, Pt and Au developed by Jacobsen, Stoltze, and Norskov (1996) v000 |
| EMT_Asap_Standard_JacobsenStoltzeNorskov_1996_Cu__MO_396616545191_001 | EMT potential for Cu developed by Jacobsen, Stoltze, and Norskov (1996) v000 |
| EMT_Asap_Standard_Jacobsen_Stoltze_Norskov_AlAgAuCuNiPdPt__MO_118428466217_002 | Standard Effective Medium Theory potential for face-centered cubic metals as implemented in ASE/Asap. |
| LJ_ElliottAkerson_2015_Universal__MO_959249795837_003 | Efficient 'universal' shifted Lennard-Jones model for all KIM API supported species developed by Elliott and Akerson (2015) v003 |
| MEAM_2NN_Fe_to_Ga__MO_145522277939_001 | Model parameterization of 2NN MEAM model |
| Morse_Shifted_GirifalcoWeizer_1959HighCutoff_Cu__MO_151002396060_002 | Morse potential (shifted) for Cu by Girifalco and Weizer (1959) using a high-accuracy cutoff distance v002 |
| Morse_Shifted_GirifalcoWeizer_1959LowCutoff_Cu__MO_673777079812_002 | Morse potential (shifted) for Cu by Girifalco and Weizer (1959) using a low-accuracy cutoff distance v002 |
| Morse_Shifted_GirifalcoWeizer_1959MedCutoff_Cu__MO_173787283511_002 | Morse potential (shifted) for Cu by Girifalco and Weizer (1959) using a medium-accuracy cutoff distance v002 |
| Pair_Morse_Modified_MacDonaldMacDonald_Cu__MO_034823476734_000 | Modified Morse pair potential for copper due to MacDonald and MacDonald |

# Models on openkim.org

## EAM_NN_Johnson_1988_Cu__MO_887933271505_002

| | |
|---|---|
| **Title** | EAM Potential (analytical nearest-neighbor) for Cu developed by Johnson (1988) v002 |
| **Description** | Analytical nearest-neighbor EAM model for Cu by Johnson |
| **Species** | Cu |
| **Contributor** | Ryan |
| **Maintainer** | Ryan |
| **Author** | Ryan S. Elliott |
| **Publication Year** | 2018 |
| **Item Citation** | This Model originally published in [1] is archived in OpenKIM [2-4].<br><br>[1] Johnson RA. Analytic nearest-neighbor model for fcc metals. Physical Review B. 1988Mar;37(8):3924–31. doi:10.1103/PhysRevB.37.3924<br><br>[2] Elliott RS. EAM Potential (analytical nearest-neighbor) for Cu developed by Johnson (1988) v002. OpenKIM; 2018. doi:10.25950/3ccd9f3b<br><br>[3] Tadmor EB, Elliott RS, Sethna JP, Miller RE, Becker CA. The potential of atomistic simulations and the Knowledgebase of Interatomic Models. JOM. 2011;63(7):17. doi:10.1007/s11837-011-0102-6<br><br>[4] Elliott RS, Tadmor EB. Knowledgebase of Interatomic Models (KIM) Application Programming Interface (API). OpenKIM; 2011. doi:10.25950/ff8f563a<br><br>Click here to download the above citation in BibTeX format. |
| **Short KIM ID** | MO_887933271505_002 |
| **Extended KIM ID** | EAM_NN_Johnson_1988_Cu__MO_887933271505_002 |
| **DOI** | 10.25950/3ccd9f3b<br>https://doi.org/10.25950/3ccd9f3b<br>https://search.datacite.org/works/10.25950/3ccd9f3b |
| **KIM Item Type** | Portable Model |
| **KIM API Version** | 2.0 |
| **Programming Language(s)** | 100.00% C |
| **Previous Version** | EAM_NN_Johnson_1988_Cu__MO_887933271505_001 |

# Models on openkim.org

Further down the model page for
EAM_NN_Johnson_1988_Cu__MO_887933271505_002

## Verification Check Dashboard

(Click here to learn more about Verification Checks)

| Grade | Name | Category | Brief Description | Full Results | Aux File(s) |
|---|---|---|---|---|---|
| P | **vc-species-supported-as-stated** | mandatory | The model supports all species it claims to support; see full description. | Results | Files |
| P | **vc-periodicity-support** | mandatory | Periodic boundary conditions are handled correctly; see full description. | Results | Files |
| P | **vc-permutation-symmetry** | mandatory | Total energy and forces are unchanged when swapping atoms of the same species; see full description. | Results | Files |
| A | **vc-forces-numerical-derivative** | consistency | Forces computed by the model agree with numerical derivatives of the energy; see full description. | Results | Files |
| F | **vc-dimer-continuity-c1** | informational | The energy versus separation relation of a pair of atoms is C1 continuous (i.e. the function and its first derivative are continuous); see full description. | Results | Files |
| P | **vc-objectivity** | informational | Total energy is unchanged and forces transform correctly under rigid-body translation and rotation; see full description. | Results | Files |
| P | **vc-inversion-symmetry** | informational | Total energy is unchanged and forces change sign when inverting a configuration through the origin; see full description. | Results | Files |
| P | **vc-memory-leak** | informational | The model code does not have memory leaks (i.e. it releases all allocated memory at the end); see full description. | Results | Files |
| P | **vc-thread-safe** | mandatory | The model returns the same energy and forces when computed in serial and when using parallel threads for a set of configurations. Note that this is not a guarantee of thread safety; see full description. | Results | Files |

# Models on openkim.org

Further down the model page for
EAM_NN_Johnson_1988_Cu__MO_887933271505_002

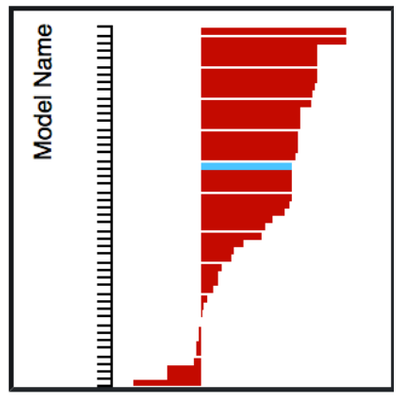
## Tests

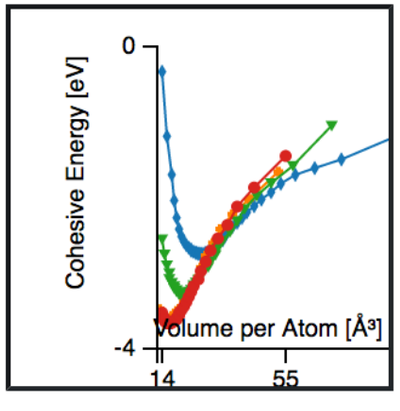
**ElasticConstantsCubic__TD_011862047401_004**

Computes the cubic elastic constants for some common crystal types (fcc, bcc, sc) by calculating the hessian of the energy density with respect to strain. An estimate of the error associated with the numerical differentiation performed is reported.

| Test | Test Results | Link to Test Results page | Benchmark time |
|---|---|---|---|
| ElasticConstantsCubic_bcc_Cu__TE_091603841600_004 | expand | view | 2602 |
| ElasticConstantsCubic_fcc_Cu__TE_188557531340_004 | expand | view | 3665 |
| ElasticConstantsCubic_sc_Cu__TE_319353354686_004 | expand | view | 3079 |

| | Model | Lattice Constant [Å] | Cohesive Energy [eV] | c11 [GPa] | c12 [GPa] | c44 [GPa] |
|---|---|---|---|---|---|---|
| bcc Expand | EAM_NN_Johnson_1988_Cu__MO_887933271505_002 | 2.8593961074900003 | 3.6063831577047547 | 146.26088738200002 | 137.952181442 | 91.93678176490002 |
| diamond Expand | EAM_NN_Johnson_1988_Cu__MO_887933271505_002 | 5.450421601530001 | 2.4241832490740016 | N/A | N/A | N/A |
| fcc Expand | EAM_NN_Johnson_1988_Cu__MO_887933271505_002 | 3.61472985148 | 3.540000123312368 | 184.172808464 | 115.324864335 | 68.8519693905 |
| sc Expand | EAM_NN_Johnson_1988_Cu__MO_887933271505_002 | 2.3724498152700004 | 3.263473577969841 | 270.847253148 | 24.4996165814 | -17.5854303931 |



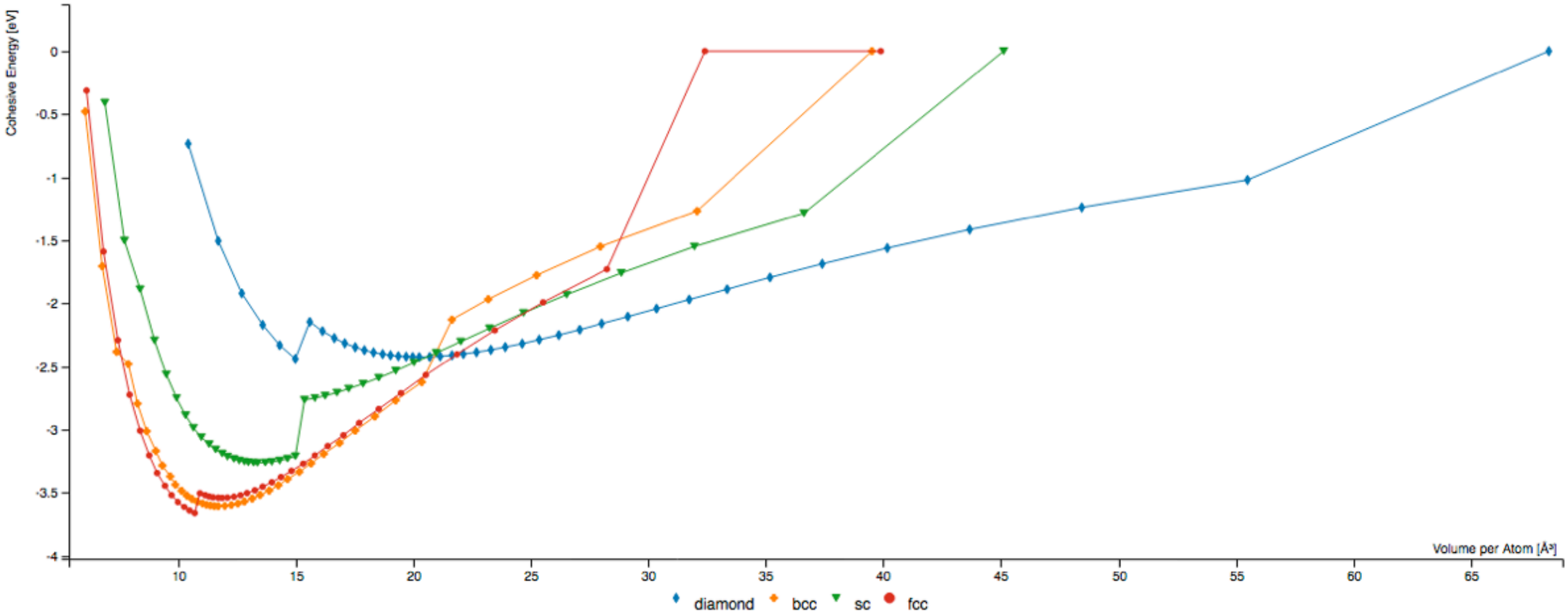



# Models on openkim.org

OpenKIM

## Model: EAM_NN_Johnson_1988_Cu__MO_887933271505_002
## Species: Cu

This graph shows the cohesive energy versus volume-per-atom for the current model for four mono-atomic cubic phases (body-centered cubic (bcc), face-centered cubic (fcc), simple cubic (sc), and diamond). The curve with the lowest minimum is the ground state of the crystal if stable. (The crystal structure is enforced in these calculations, so the phase may not be stable.) The curves below are for the species specified above.



# KIM-Compliant Codes

ase

nanoHUB

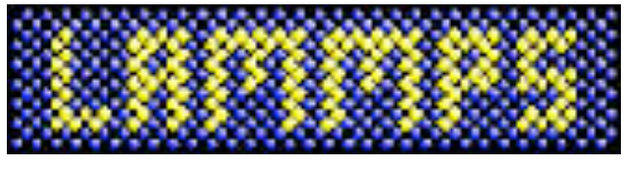
Asap

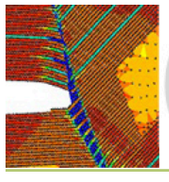
potfit

DL_POLY

GULP

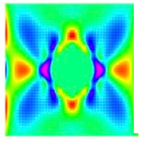
KLiFF





QuasiContinuum Method
the original source for information, publications, and downloads

libAtoms + QUIP



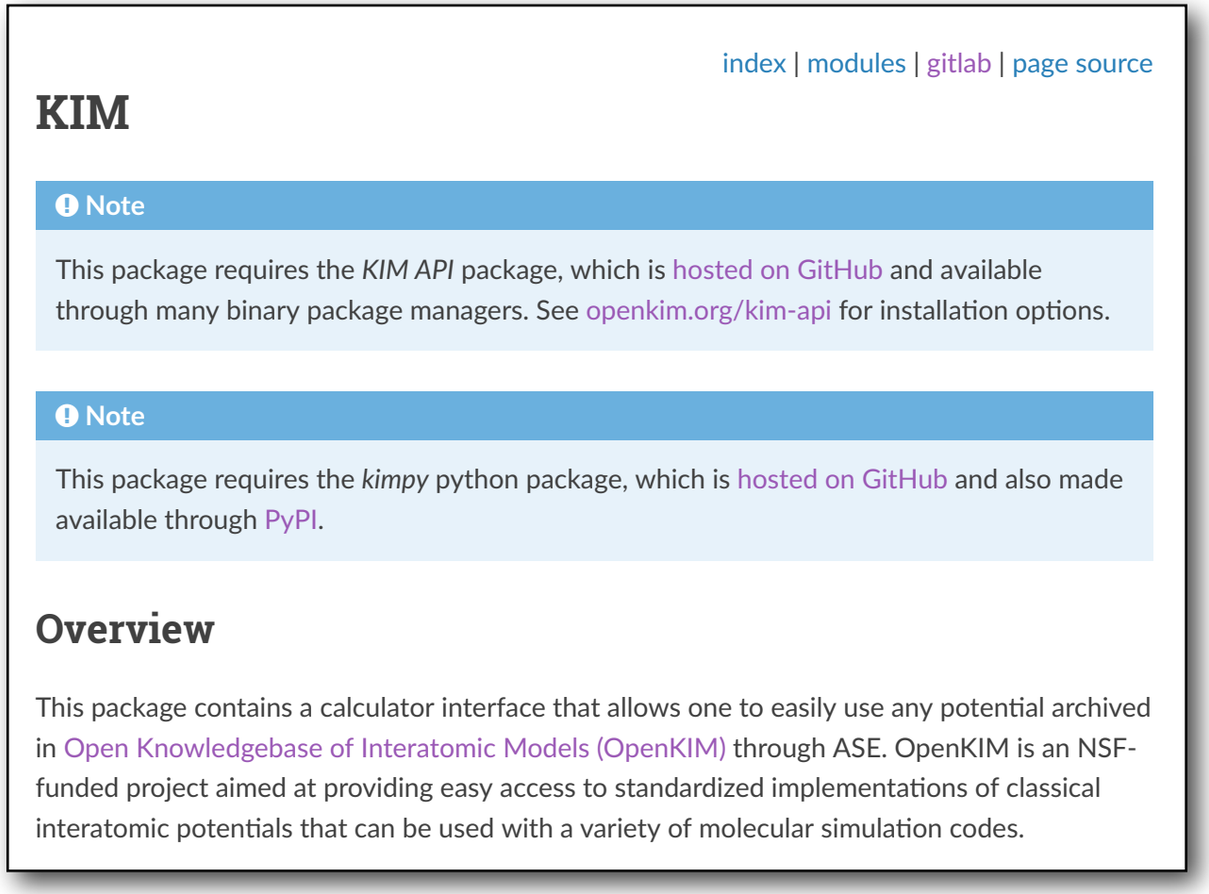
MDStressLab

# Using KIM IMs with ASE

- The latest release of ASE provides full support for using KIM IMs

  This includes:

  - Easy installation from source, binary or Conda
  - Full support for KIM Portable Models (PMs) and Simulator Models (SMs) through a **KIM Calculator**
  - Ability to query IM property predictions from openkim.org to be used in setting up and analyzing ASE simulations



index | modules | gitlab | page source

## KIM

**Note**

This package requires the *KIM API* package, which is hosted on GitHub and available through many binary package managers. See openkim.org/kim-api for installation options.

**Note**

This package requires the *kimpy* python package, which is hosted on GitHub and also made available through PyPI.

### Overview

This package contains a calculator interface that allows one to easily use any potential archived in Open Knowledgebase of Interatomic Models (OpenKIM) through ASE. OpenKIM is an NSF-funded project aimed at providing easy access to standardized implementations of classical interatomic potentials that can be used with a variety of molecular simulation codes.

See details at

https://wiki.fysik.dtu.dk/ase/dev/ase/calculators/kim.html

# Installing ASE with KIM Support

- Using ASE with KIM requires installation of the KIM API, KIM models library, and KIM queries.

**Direct Binary Installation**

```
Install 'kim-api' and 'openkim-models' via package manager, e.g. brew, apt-get
(see https://openkim.org/kim-api/)

$ pip install --user kimpy (requires gcc)
$ pip install git+https://github.com/openkim/kim-python-utils (optional for queries)
$ pip install git+https://gitlab.com/ase/ase.git
```

**Installation using Conda**

```
$ conda create --name ase && conda activate ase (create Conda environment)
$ conda install -c conda-forge kim-api openkim-models kimpy (install KIM)
$ conda install -c conda-forge matplotlib (matplotlib required by ASE)
$ pip install git+https://gitlab.com/ase/ase.git (install ASE)
$ pip install git+https://github.com/openkim/kim-python-utils (optional for queries)
```

# Using a KIM IM in ASE

- KIM IMs are used in ASE using the KIM Calculator.

Vim

```
#!/usr/bin/env python3
"""
Compute the cohesive energy and pressure of an FCC Al crystal with a0=4.05 for KIM
model EAM_Dynamo_ErcolessiAdams_1994_Al__MO_123629422045_005
"""
from ase.calculators.kim import KIM
from ase.lattice.cubic import FaceCenteredCubic
from ase.units import GPa

# Set up crystal and calculator
a0 = 4.05  # experimental lattice constant
atoms = FaceCenteredCubic("Al", latticeconstant=a0)
calc = KIM("EAM_Dynamo_ErcolessiAdams_1994_Al__MO_123629422045_005")
atoms.set_calculator(calc)

# Compute energy/pressure
ecoh = -atoms.get_potential_energy() / len(atoms)
stress = atoms.get_stress()
pressure_MPa = (-sum(stress[:3]) / 3.0) * 1e3 / GPa

print("Computed cohesive energy of {:.3f} eV/atom (experiment: 3.39 eV/atom)".format(ecoh))
print("Computed pressure of {} MPa".format(pressure_MPa))
```

Load the KIM calculator

Select KIM model and attach to atomic configuration

# Using a KIM IM in ASE

- Running this gives:

```
(ase) root@6ddb06015b8e:~/ase_workshop_examples# python ex1.py
Computed cohesive energy of 3.359 eV/atom (experiment: 3.39 eV/atom)
Computed pressure of -1049.6852893566931 MPa
(ase) root@6ddb06015b8e:~/ase_workshop_examples#
```

- KIM provides full citation information including DOIs for all models:

DataCite
FIND, ACCESS, AND REUSE DATA

This Model originally published in [1] is archived in OpenKIM [2-5].

[1] Ercolessi F, Adams JB. Interatomic Potentials from First-Principles Calculations: The Force-Matching Method. Europhysics Letters. 1994;26(8):583. doi:10.1209/0295-5075/26/8/005

Scientific citation

[2] Elliott RS. EAM potential (LAMMPS cubic hermite tabulation) for Al developed by Ercolessi and Adams (1994) v005. OpenKIM; 2018. doi:10.25950/7cd2a6ab

[3] Elliott RS. EAM Model Driver for tabulated potentials with cubic Hermite spline interpolation as used in LAMMPS v005. OpenKIM; 2018. doi:10.25950/68defa36

Code used in the simulation

[4] Tadmor EB, Elliott RS, Sethna JP, Miller RE, Becker CA. The potential of atomistic simulations and the Knowledgebase of Interatomic Models. JOM. 2011;63(7):17. doi:10.1007/s11837-011-0102-6

[5] Elliott RS, Tadmor EB. Knowledgebase of Interatomic Models (KIM) Application Programming Interface (API). OpenKIM; 2011. doi:10.25950/ff8f563a

KIM infrastructure

Click here to download the above citation in BibTeX format.

Full citation in BibTex format

# KIM Queries

- You may have noticed in the previous calculation that the pressure was far from zero:

```
(ase) root@6ddb06015b8e:~/ase_workshop_examples# python ex1.py
Computed cohesive energy of 3.359 eV/atom (experiment: 3.39 eV/atom)
Computed pressure of -1049.6852893566931 MPa
(ase) root@6ddb06015b8e:~/ase_workshop_examples#
```

This is because the lattice parameter used (a0=4.05 Å) is <u>not</u> the equilibrium lattice constant predicted by the Ercolessi-Adams potential for fcc Al.

- A mechanism is provided for querying properties (like a0) from openkim.org:

```
model = "EAM_Dynamo_ErcolessiAdams_1994_Al__MO_123629422045_005"

# Perform query to get lattice constant for this model
a0 = get_lattice_constant_cubic([model], ["fcc"], ["Al"], ["angstrom"])[0]
```

This sets a0=4.032082033157349 (obtained from openkim.org) which gives:

```
Computed cohesive energy of 3.360 eV/atom (experiment: 3.39 eV/atom)
Computed pressure of -0.0013119052397128333 MPa
```

# KIM Queries for Robust Computations

- KIM queries provide a powerful mechanism for writing ASE scripts that work for any interatomic model without having to repeat complex and expensive computations.

**EXAMPLE:** MD simulation in the *NPT* ensemble of an fcc crystal

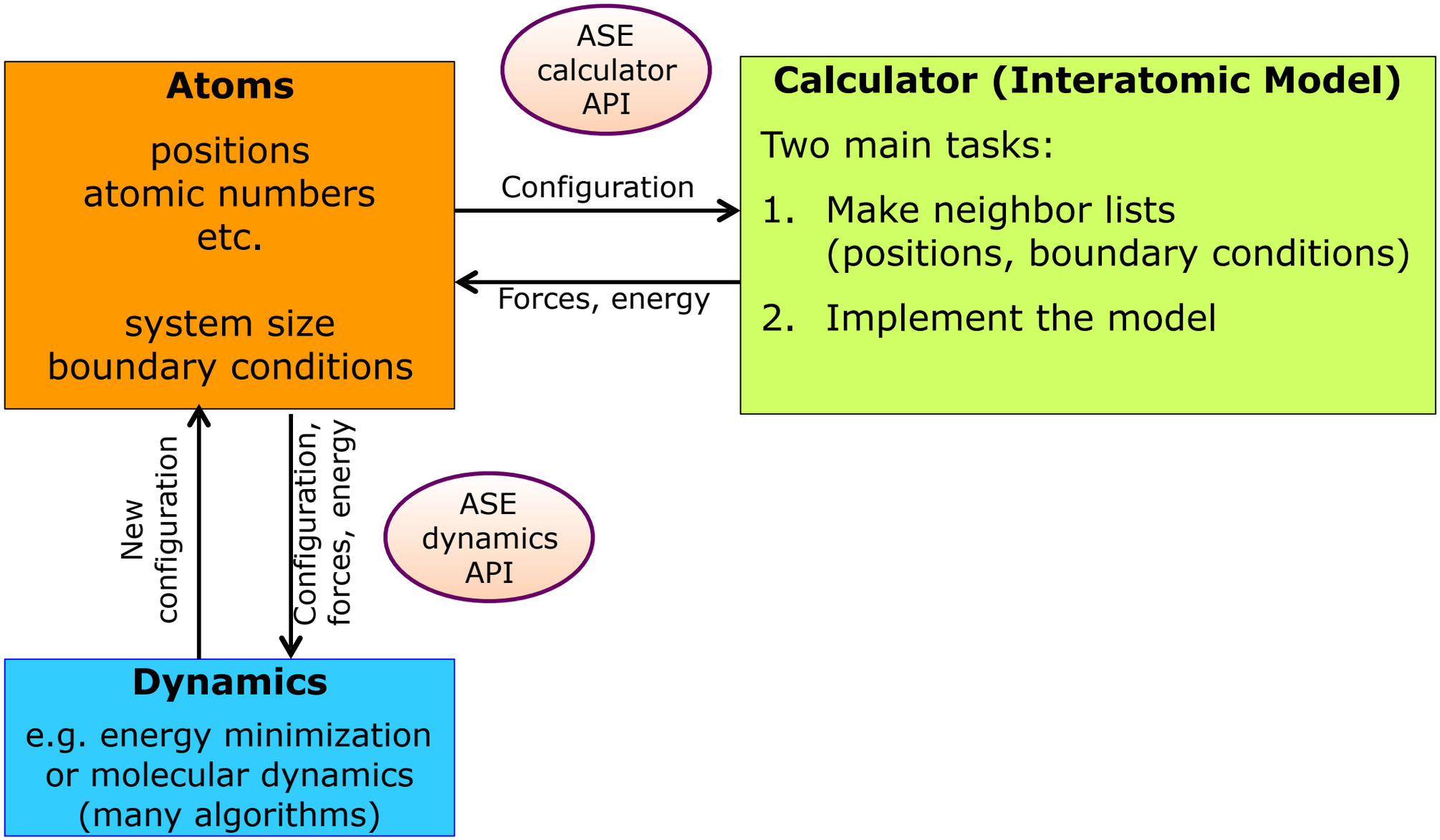
The following algorithm will work robustly for <u>any</u> interatomic model

- Query openkim.org to obtain the following properties for the fcc structure:
  - $a_0$ : equilibrium lattice constant
  - $\alpha$ : thermal expansion coefficient
  - $B$ : bulk modulus
- Construct fcc crystal for the scaled lattice constant: $a = a_0(1 + \alpha T)$
- Equilibrate until the cell volume fluctuations converge to the expected value from statistical mechanics:

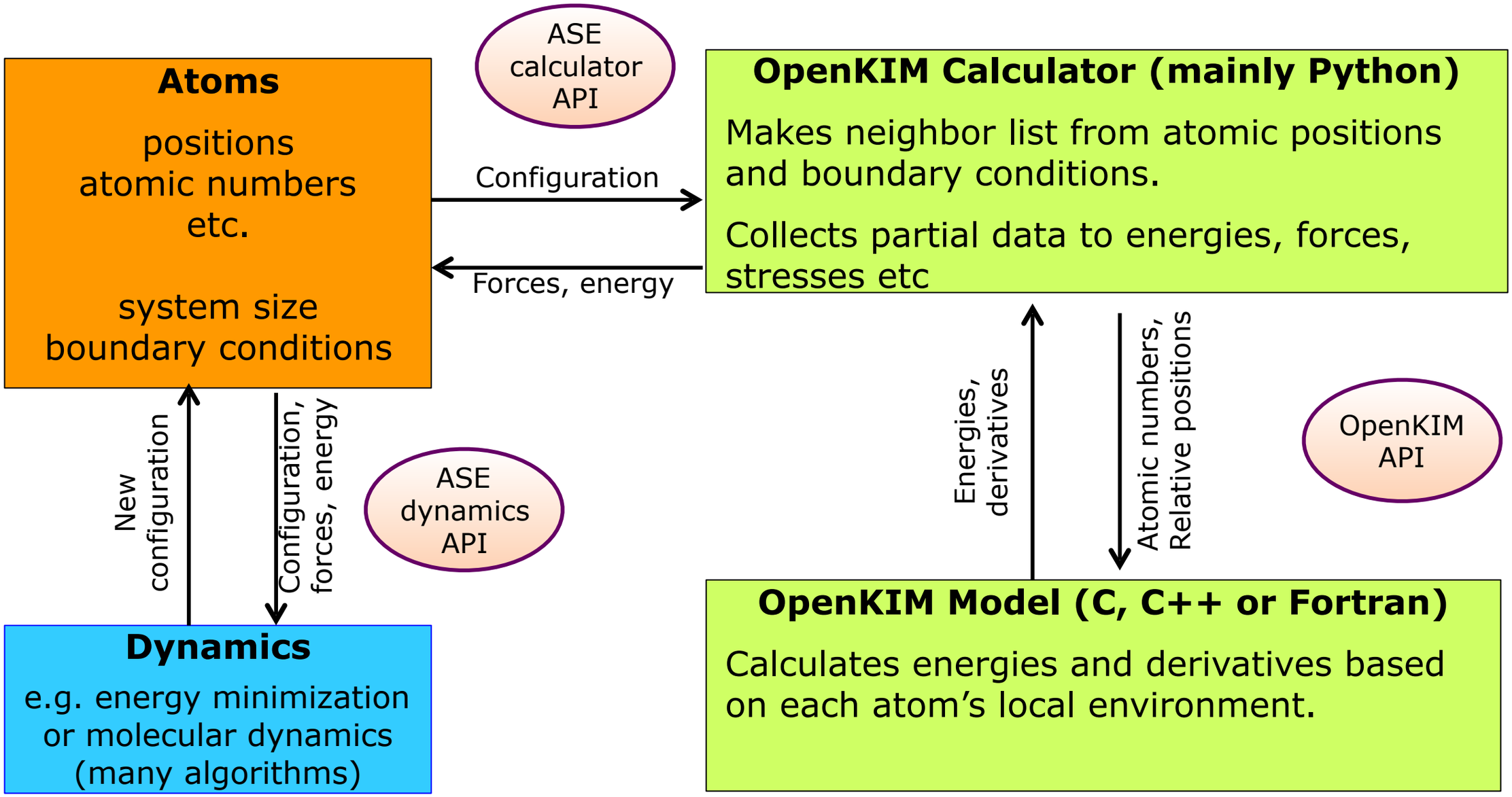
$$\frac{\Delta V}{\sqrt{V}} = \sqrt{\frac{K_{\mathrm{B}} T}{B}}$$

- Proceed with simulation...

# The ASE calculator API and the OpenKIM API



# The ASE calculator API and the OpenKIM API



# At least two OpenKIM calculators in ASE

**KIM (multiple calculators)**

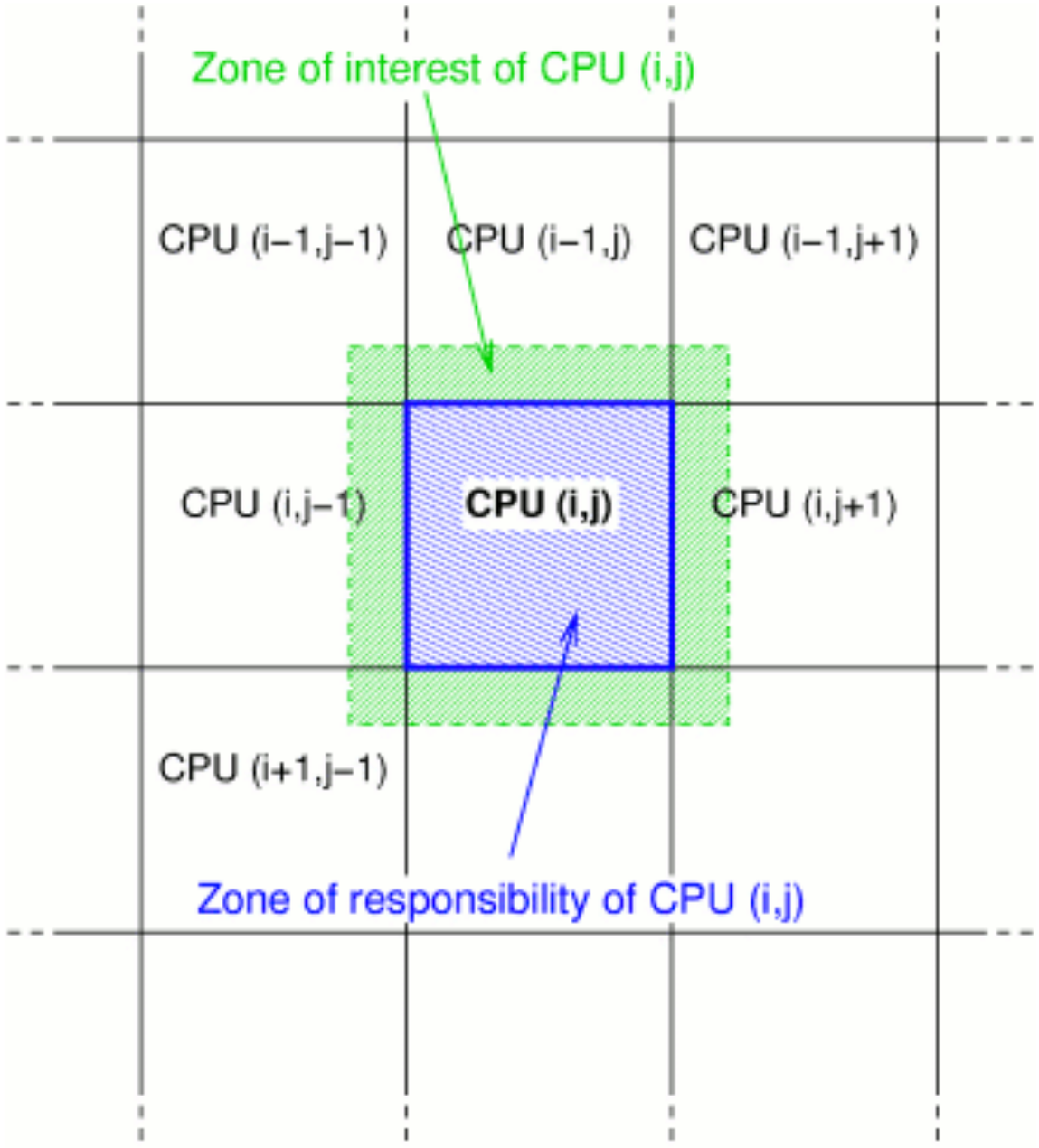
- An ASE calculator
- Supports both Portable Models and Simulator Models (if LAMMPS installed)
- Easy to use for serial simulations
  –
- Part of ASE, pure Python on top of kimpy

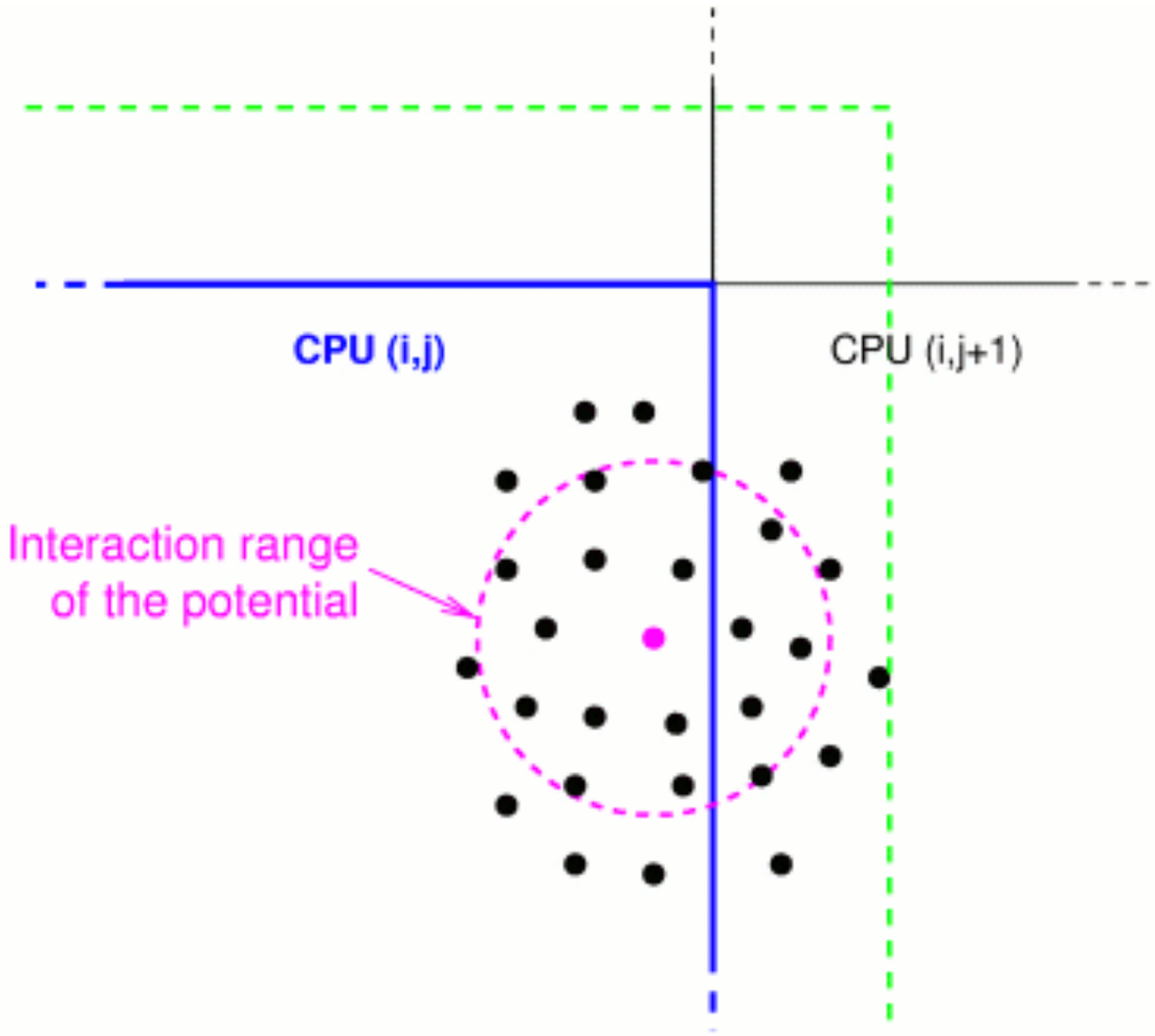
**ASAP**

- An ASE calculator
- Supports Portable Models
- An ASE extension for large-scale parallel simulations
  – Domain decomposition
- Separate project: C++, Python

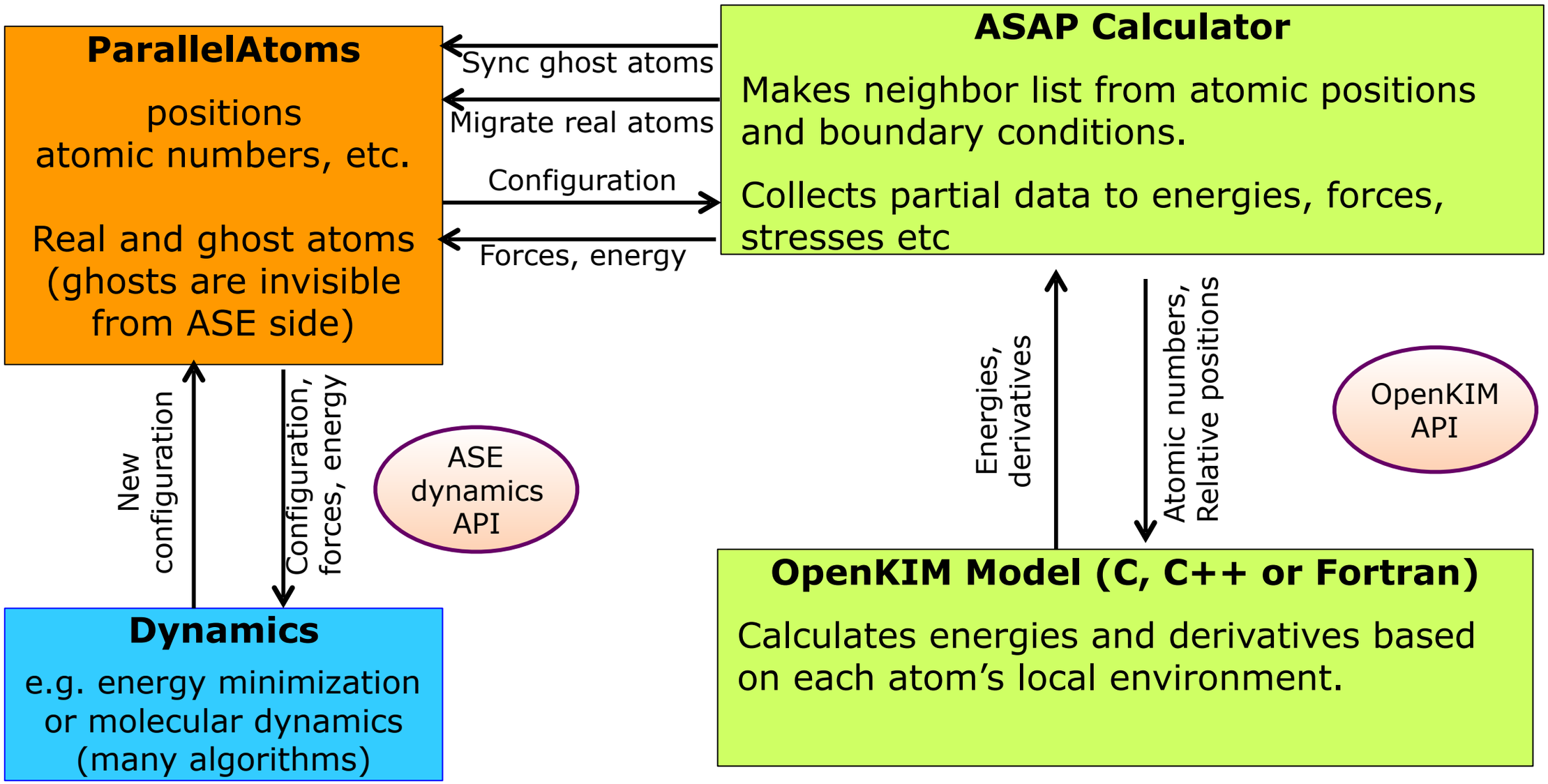
# Massively parallel molecular dynamics: Domain decomposition

Simulation with millions of atoms.



# Asap must mess with ASE objects!

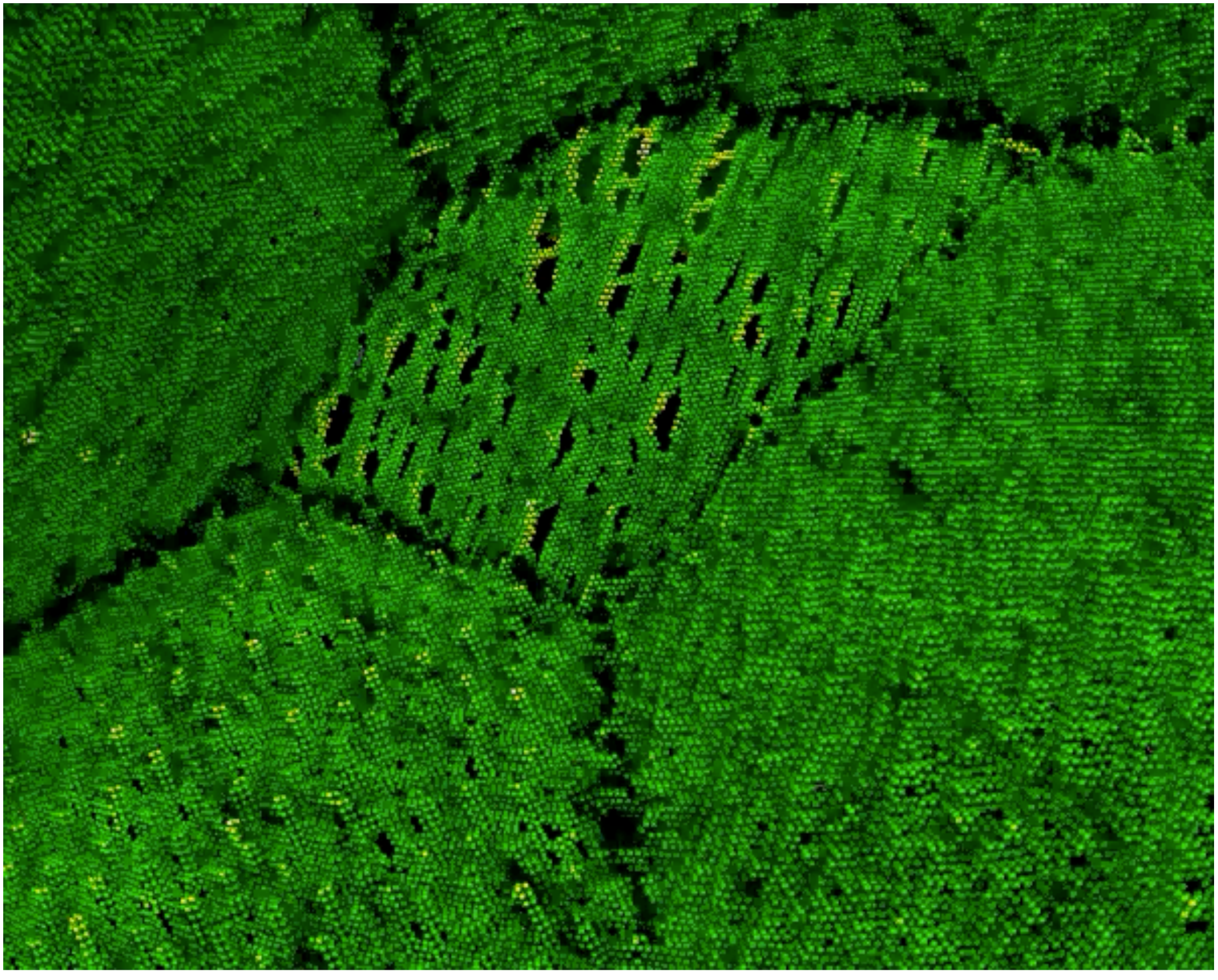


# Large-scale molecular dynamics

Dislocation motion in polycrystalline copper

100 million atoms.
Atoms in FCC structure are hidden in video

Ran on 100 cores in 2003.



Schiøtz and Jacobsen, *Science* **301**, 135 (2003).

# Conclusions

**The Knowledgebase of Interatomic Models (KIM) provides:**

- A curated **repository** of Interatomic Models (IMs) with version control and DOIs for easy referencing.
- A **standard API** so molecular simulation codes can use these IMs.
- A set of **verification and evaluation tests** to aid you in selecting an appropriate IM.
- **Easily accessible from ASE** through ase.calculators.kim.KIM() for serial simulations and from Asap through asap3.OpenKIMcalculator() for massively parallel simulations.

**OpenKIM.org**

"All models are wrong, but some are useful."

*George E. P. Box*