A "How-To" Tutorial on Using OpenKIM with LAMMPS

Ryan S. Elliott, Ellad B. Tadmor, and Daniel S. Karls
Department of Aerospace Engineering and Mechanics
University of Minnesota

Key Collaborators:

- Yaser Afshar (U. Minnesota)
- Noam Bernstein (NRL)
- George Karypis (U. Minnesota)
- Yonatan Kurniawan (BYU)
- Mark Transtrum (BYU)
- Mingian Wen (LBL)

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

How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

LAMMPS Binary (w/ kim-api)

2. Install LAMMPS

You can download LAMMPS as an executable or as source code.

With source code, you also have to build LAMMPS. But you have more flexibility as to what features to include or exclude in the build. If you plan to modify or extend LAMMPS, then you need the source code.

- 2.1. Download an executable for Linux
- 2.2. Download an executable for Mac
- 2.3. Download an executable for Windows

2.2. Download an executable for Mac

LAMMPS can be downloaded, built, and configured for OS X on a Mac with Homebrew. The following LAMMPS packages are unavailable at this time because of additional needs not yet met: GPU, KOKKOS, LATTE, MSCG, MESSAGE, MPIIO POEMS VORONOI.

After installing Homebrew, you can install LAMMPS on your system with the following commands:

```
% brew install lammps
```

This will install the executables “lammps_serial” and “lammps_mpi”, as well as the LAMMPS “doc”, “potentials”, “tools”, “bench”, and “examples” directories.

The LAMMPS binary is built with the KIM package which results in Homebrew also installing the *kim-api* binaries when LAMMPS is installed. In order to use potentials from openkim.org, you can install the *openkim-models* package

```
% brew install openkim-models
```

LAMMPS binary: macOS with Homebrew (https://brew.sh)

- Set up homebrew following the instructions at https://brew.sh
- Then (in Terminal):

```
%
%
% brew install lammps
==> Installing dependencies for lammps: open-mpi, fftw and kim-api
==> Installing lammps dependency: open-mpi
==> Downloading https://homebrew.bintray.com/bottles/open-mpi-4.0.1_2.sierra.bottle.1.tar.gz
==> Pouring open-mpi-4.0.1_2.sierra.bottle.1.tar.gz
    /usr/local/Cellar/open-mpi/4.0.1_2: 756 files, 10.8MB
==> Installing lammps dependency: fftw
==> Downloading https://homebrew.bintray.com/bottles/fftw-3.3.8_1.sierra.bottle.tar.gz
==> Pouring fftw-3.3.8_1.sierra.bottle.tar.gz
    /usr/local/Cellar/fftw/3.3.8_1: 73 files, 14.8MB
==> Installing lammps dependency: kim-api
==> Downloading https://homebrew.bintray.com/bottles/kim-api-2.1.2.sierra.bottle.tar.gz
==> Pouring kim-api-2.1.2.sierra.bottle.tar.gz
==> Installing lammps
==> Downloading https://homebrew.bintray.com/bottles/lammps-2019-08-07.sierra.bottle.tar.gz
==> Pouring lammps-2019-08-07.sierra.bottle.tar.gz
%
%
% brew install openkim-models
==> Downloading https://homebrew.bintray.com/bottles/openkim-models-2019-07-25.sierra.bottle.tar.gz
==> Pouring openkim-models-2019-07-25.sierra.bottle.tar.gz
    /usr/local/Cellar/openkim-models/2019-07-25: 415 files, 223.9MB
%
%
```

- Other binary package installations are similar

https://openkim.org/doc/usage/obtaining-models/

Many other systems are supported by OpenKIM, and we continue to add more.

Let us know if your favorite system is not on the list! (Windows support is "in progress")

Obtaining KIM Models

Documentation / Using KIM Content / Obtaining KIM Models

In order to use KIM Portable Models (PMs), which may depend on Model Drivers (MDs), or Simulator Models (SMs) on your system, you must install the KIM application programming interface (API) along with the specific models that you are interested in using. There are two options for doing so:

- Installing from pre-built binaries
  - Fedora linux
  - Homebrew macOS
  - openSUSE linux
  - Ubuntu linux
  - FreeBSD Ports
  - Spack
- Installing from source
  - KIM API
  - Models

Installing from Pre-built Binaries

The easiest option for obtaining the KIM API and KIM models is to install the `openkim-models` pre-compiled binary package for your preferred operating system or package manager. When you install the `openkim-models` package, the package manager will automatically install the KIM API shared library as a required dependency.

The table to the right lists the systems for which pre-compiled binaries are available. For each system, the currently available version (release date) of the `openkim-models` package is listed. Green indicates the most current version available, whereas red indicates an older version. Binary packages are released on a regular basis, however if the model you require is not contained in the current version, you can install a model from source as explained below.

Packaging status	
EPEL 7	2019.03.31
Fedora 28	2019.03.31
Fedora 29	2019.03.31
Fedora 30	2019.03.31
Fedora Rawhide	2019.03.31
FreeBSD Ports	2019.07.25
Homebrew	2019-07-25
Linuxbrew	2019-07-25
openSUSE Tumbleweed	2019.03.31
openSUSE Science Tumbleweed	2019.07.25

The commands necessary to install from binary on the various supported systems are:

How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

LAMMPS from source:

3.1. Build LAMMPS with CMake

This page is a short summary of how to use CMake to build LAMMPS. Details on CMake variables that enable specific LAMMPS build options are given on the pages linked to from the Build doc page.

Richard Berger (Temple U) has also written a more comprehensive guide for how to use CMake to build LAMMPS. If you are new to CMake it is a good place to start.

Building LAMMPS with CMake is a two-step process. First you use CMake to create a build environment in a new directory. On Linux systems, this will be based on makefiles for use with make. Then you use the make command to build LAMMPS, which uses the created Makefile(s). Example:

```
cd lammps                       # change to the LAMMPS distribution directory
mkdir build; cd build           # create a new directory (folder) for build
cmake [options ...] ../cmake    # configuration with (command-line) cmake
make                            # compilation
```


CMake variables:

```
-D PKG_NAME=value          # yes or no (default)
```

Examples:

```
-D PKG_MANYBODY=yes
-D PKG_USER-INTEL=yes
```

```
% mkdir lammps-setup
% cd lammps-setup

% wget -q https://lammps.sandia.gov/tars/lammps-7Aug19.tar.gz

% tar zxf lammps-7Aug19.tar.gz

% mkdir build

% ls
build        lammps-7Aug19        lammps-7Aug19.tar.gz

% cd build
% export CC=gcc-8 CXX=g++-8 FC=gfortran-8

% cmake ../lammps-7Aug19/cmake/ -DCMAKE_BUILD_TYPE=Release -DPKG_KIM=ON -DPKG_MANYBODY=ON
-- The CXX compiler identification is GNU 8.3.0
 .
 .
 .
-- Generating lmpinstalledpkgs.h...
-- Building package: MANYBODY
-- Building package: KIM
-- The following packages have been found:
 * Git
 * MPI
 * OpenMP
 * JPEG
 * PNG
 * ZLIB
 * CURL

-- <<< Build configuration >>>
   Build type       Release
   Install path     ~/.local
   Compilers and Flags:
   C++ Compiler     /usr/local/bin/g++-8
       Type         GNU
       Version      8.3.0
   C++ Flags         -fopenmp -O3 -DNDEBUG
   Defines          LAMMPS_SMALLBIG;LAMMPS_MEMALIGN=64;LAMMPS_JP
-- C Compiler /usr/local/bin/gcc-8
     Type     GNU
     Version  8.3.0
     C Flags    -O3 -DNDEBUG
-- Static libraries
-- Link libraries: /usr/local/Cellar/open-mpi/4.0.1_2/lib/libmpi
-- Using mpi with headers in /usr/local/Cellar/open-mpi/4.0.1_2/
-- Configuring done
-- Generating done
-- Build files have been written to: ~/lammps-setup/build
```

```
% make -j2
Scanning dependencies of target gitversion
Scanning dependencies of target kim_build
[  0%] Creating directories for 'kim_build'
 .
 .
 .
[100%] Linking CXX executable lmp
[100%] Built target lmp

% source kim_build-prefix/bin/kim-api-activate
```

Install all OpenKIM Models

```
% kim-api-collections-management install system OpenKIM
Downloading..... openkim-models-2019-07-25
[  0%] Generating kimspec.edn.c
 .
 .
 .
-- Installing: ...

Success!

%
```

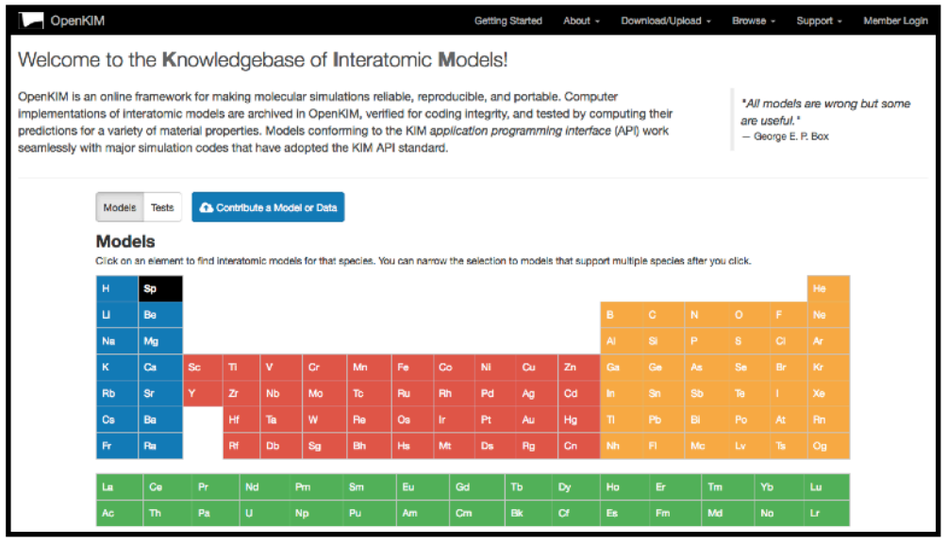
Or, install Models individually

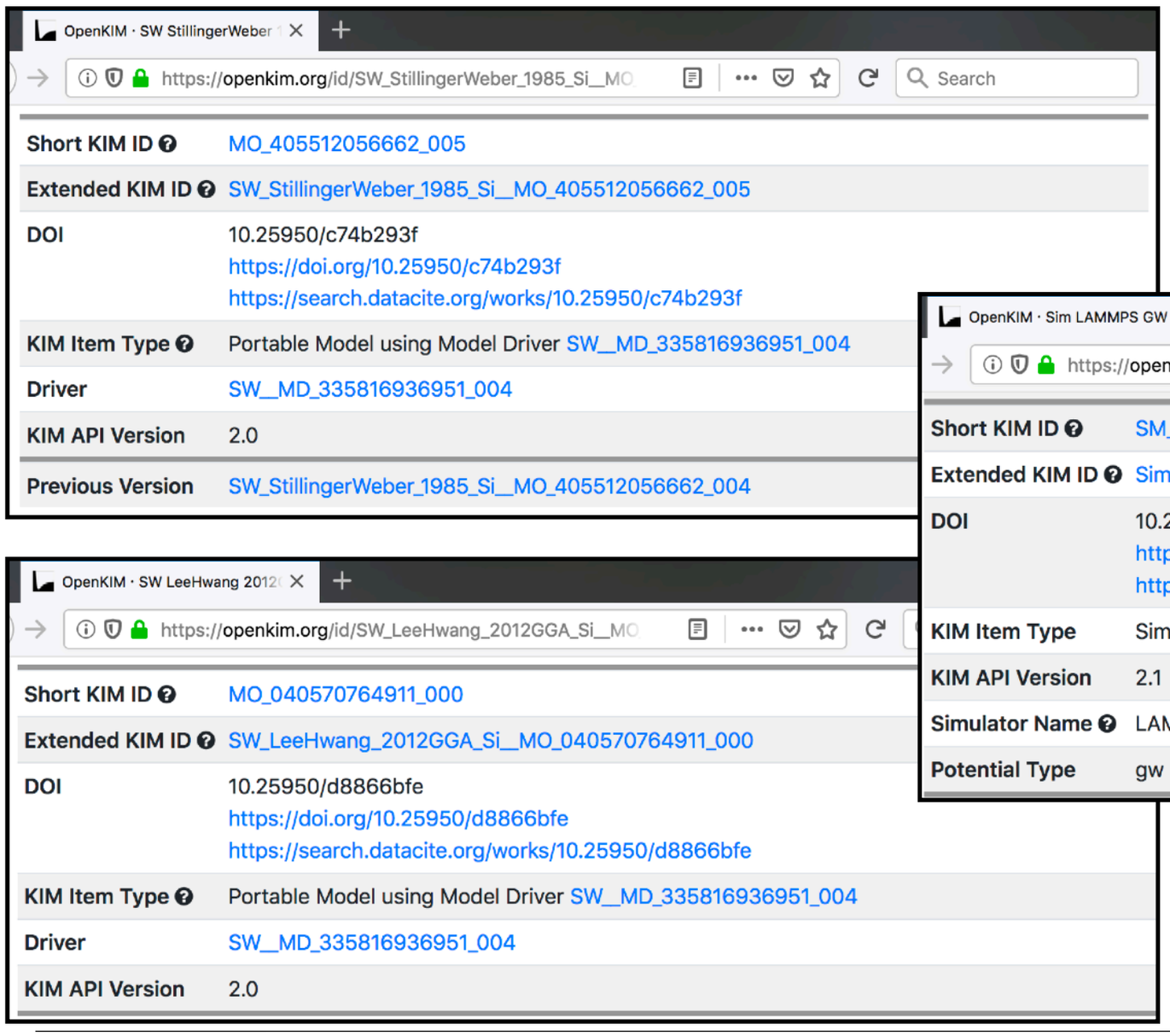
```
% kim-api-collections-management install system SW_StillingerWeber_1985_Si__MO_405512056662_005
Downloading..... SW_StillingerWeber_1985_Si__MO_405512056662_005
This item needs its driver 'SW__MD_335816936951_004', but it is not currently installed.
Trying to find it at openkim.org.
Downloading..... SW__MD_335816936951_004
[100%] Built target SW__MD_335816936951_004
Install the project...
-- Install configuration: "Release"
-- Installing: ...
[100%] Built target SW_StillingerWeber_1985_Si__MO_405512056662_005
Install the project...
-- Install configuration: "Release"
-- Installing: ...

Success!

%
```

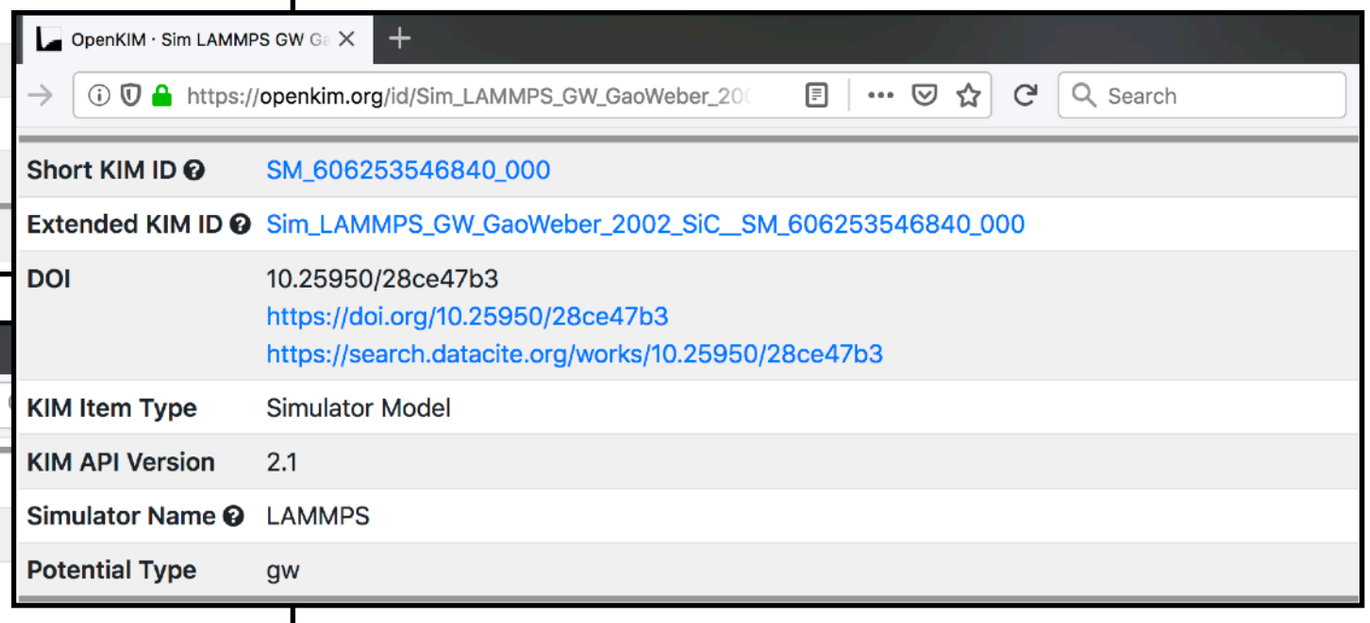
How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

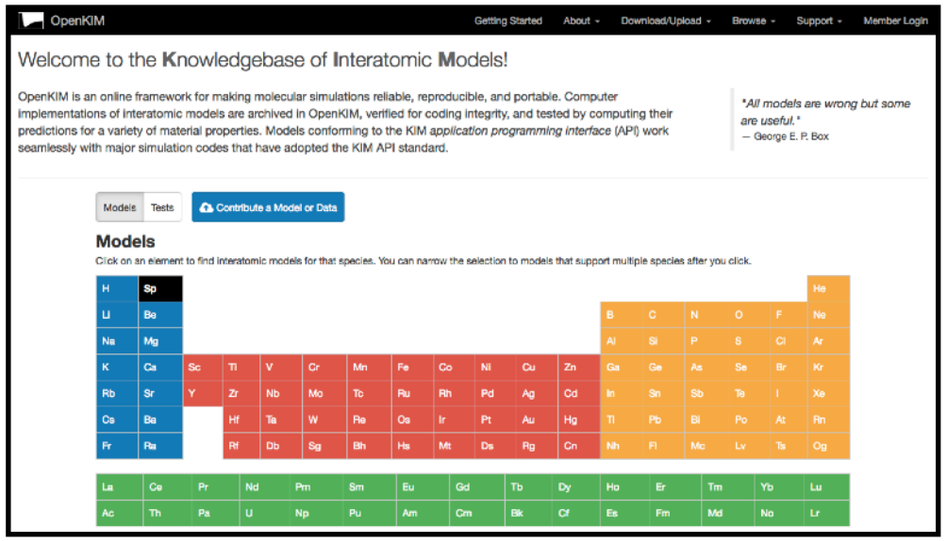
Using a KIM Model

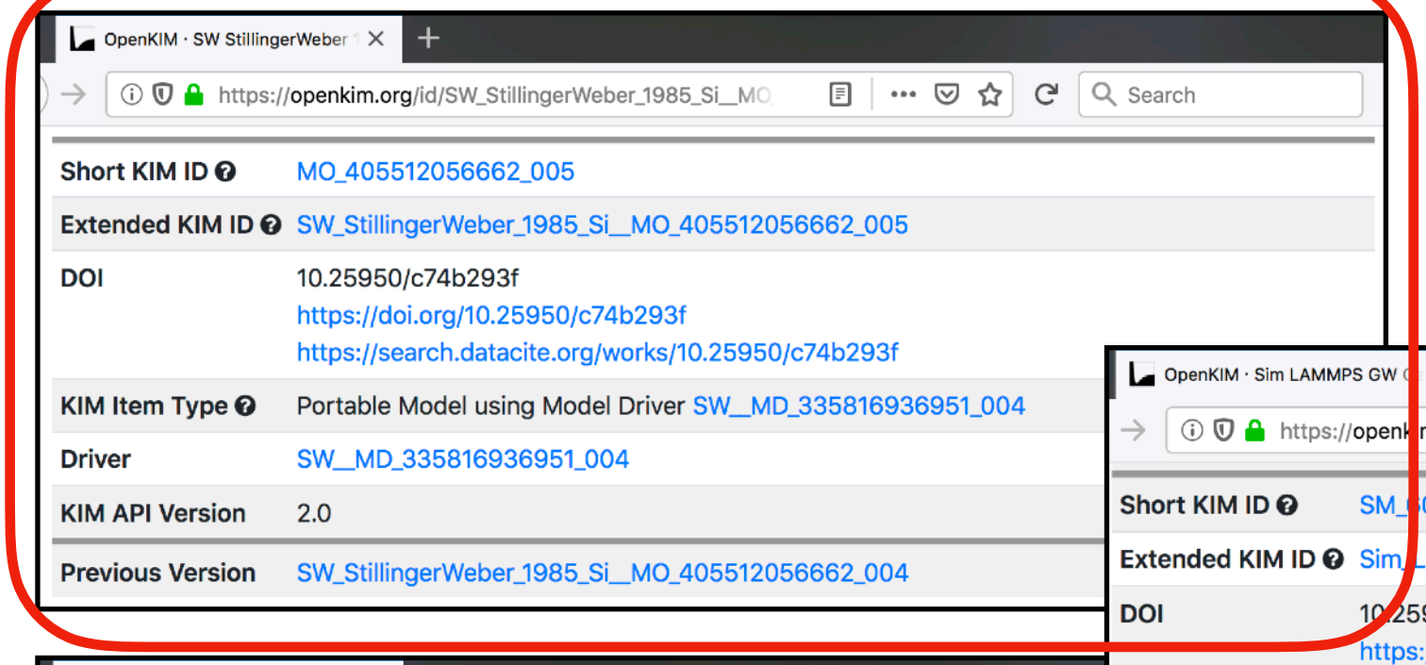


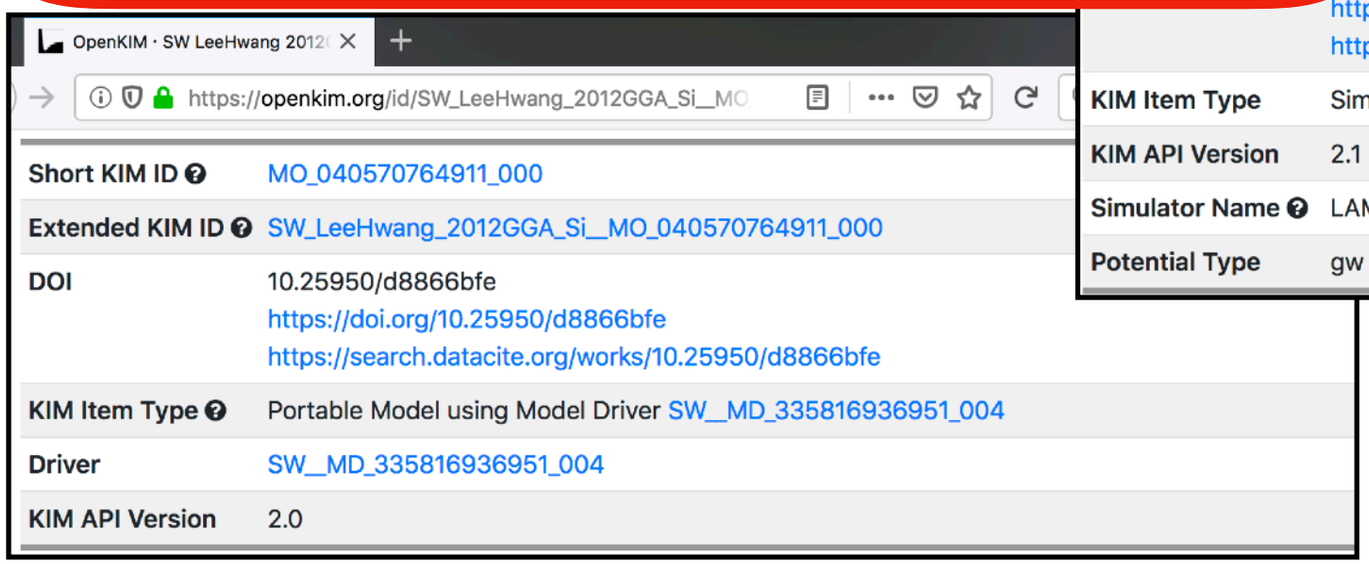




Using a KIM Model







Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                 species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

Unit consistency

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

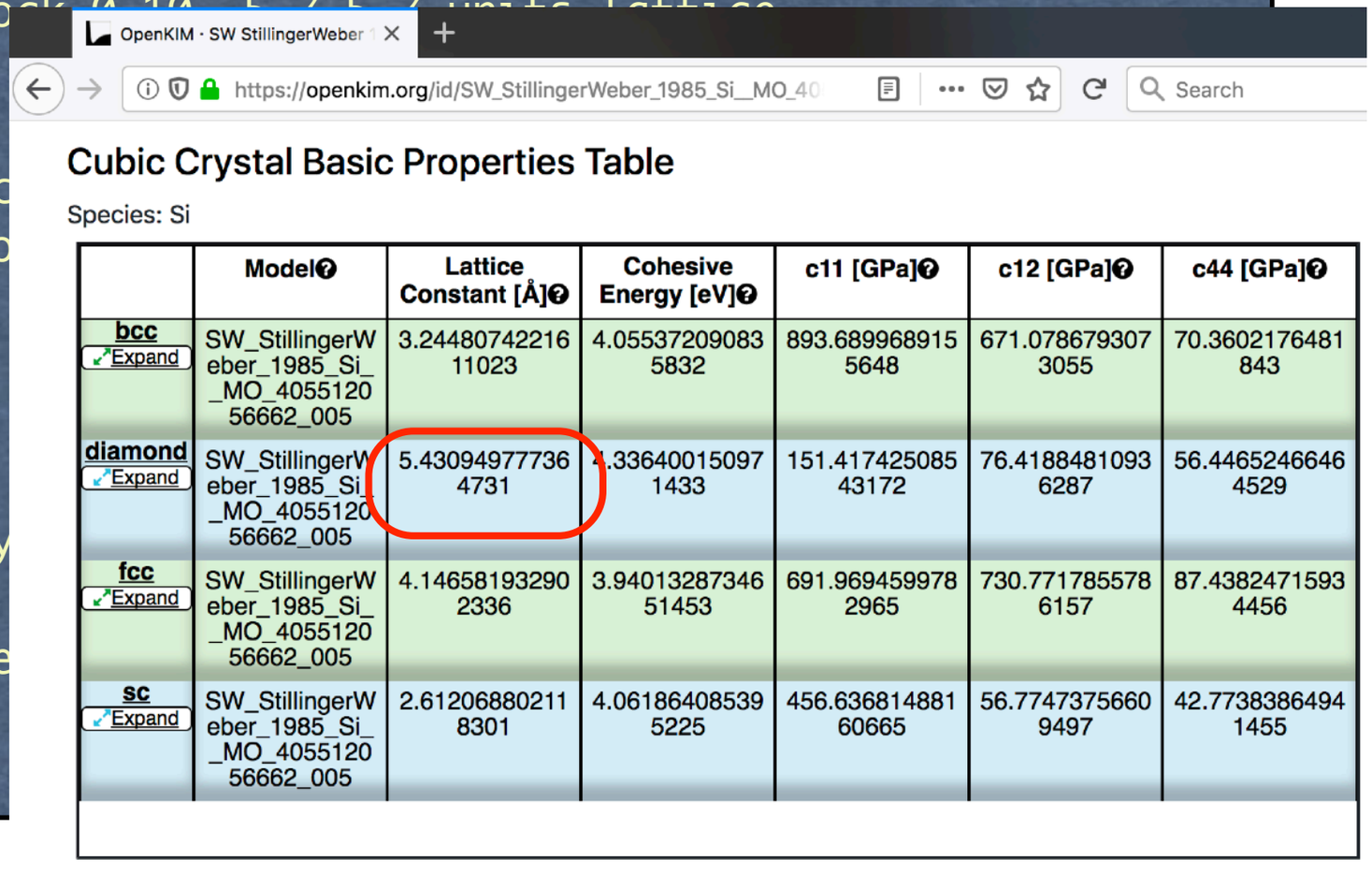
- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                        species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell blo
create_box        1 supercell
region            simbox block
region            nanowire bloc
create_atoms      1 region nano
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly
min_style cg
fix 1 all box/relax x 0.0 fixe

minimize 1.0e-16 1.0e-16 5000
```



	Model	Lattice Constant [Å]	Cohesive Energy [eV]	c11 [GPa]	c12 [GPa]	c44 [GPa]
bcc Expand	SW_StillingerWeber_1985_Si__MO_405512056662_005	3.2448074221611023	4.055372090835832	893.6899689155648	671.0786793073055	70.3602176481843
diamond Expand	SW_StillingerWeber_1985_Si__MO_405512056662_005	5.430949777364731	4.336400150971433	151.41742508543172	76.41884810936287	56.44652466464529
fcc Expand	SW_StillingerWeber_1985_Si__MO_405512056662_005	4.146581932902336	3.940132873465145 3	691.9694599782965	730.7717855786157	87.43824715934456
sc Expand	SW_StillingerWeber_1985_Si__MO_405512056662_005	2.612068802118301	4.061864085395225	456.6368148160665	56.774737566009497	42.773838649441455

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                 species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                 species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM Model with LAMMPS:

```
kim_init          SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
boundary          p p p
kim_query         a0 get_lattice_constant_cubic crystal=["diamond"] &
                                                species=["Si"] units=["angstrom"]
lattice           diamond ${a0}
region            supercell block 0 10 -5 7 5 7 units lattice
create_box        1 supercell
region            simbox block 0 10 0 2 0 2 units lattice
region            nanowire block 0 10 0 2 0 2 units lattice
create_atoms      1 region nanowire
mass              1 28.0855
kim_interactions Si

thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000
```

Using a KIM Model

- Using a KIM

```
kim_init
boundary
kim_query

lattice
region
create_box
region
region
create_ato
mass
kim_intera

thermo 10
thermo_sty
min_style
fix 1 all

minimize 1
```

```
LAMMPS (7 Aug 2019)
OMP_NUM_THREADS environment is not set. Defaulting to 1 thread. (src/comm.cpp:93)
  using 1 OpenMP thread(s) per MPI task
kim_init SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
#=== BEGIN kim-init ======================================
units metal
#=== END kim-init ========================================

boundary p p p
kim_query a0 get_lattice_constant_cubic crystal=["diamond"] species=["Si"] units=["angstrom"]
#=== BEGIN kim-query =====================================
variable a0 string 5.430949777364731
#=== END kim-query =======================================

lattice diamond ${a0}
lattice diamond 5.430949777364731
Lattice spacing in x,y,z = 5.43095 5.43095 5.43095
region           supercell block 0 10 -5 7 5 7 units lattice
create_box       1 supercell
Created orthogonal box = (0 -27.1547 27.1547) to (54.3095 38.0166 38.0166)
  1 by 1 by 1 MPI processor grid
region           simbox block 0 10 0 2 0 2 units lattice
region           nanowire block 0 10 0 2 0 2 units lattice
create_atoms     1 region nanowire
Created 0 atoms
  create_atoms CPU = 0.000153 secs
mass             1 28.0855
kim_interactions Si
#=== BEGIN kim_interactions ==============================
pair_style kim SW_StillingerWeber_1985_Si__MO_405512056662_005
pair_coeff * * Si
#=== END kim_interactions ================================


thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0
```

Using a KIM Model

- Using a KIM

```
kim_init
boundary
kim_query

lattice
region
create_box
region
region
create_ato
mass
kim_intera

thermo 10
thermo_sty
min_style
fix 1 all

minimize 1
```

```
LAMMPS (7 Aug 2019)
OMP_NUM_THREADS environment is not set. Defaulting to 1 thread. (src/comm.cpp:93)
  using 1 OpenMP thread(s) per MPI task
kim_init SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
#=== BEGIN kim-init ======================================
units metal
#=== END kim-init ========================================

boundary p p p
kim_query a0 get_lattice_constant_cubic crystal=["diamond"] species=["Si"] units=["angstrom"]
#=== BEGIN kim-query =====================================
variable a0 string 5.430949777364731
#=== END kim-query =======================================

lattice diamond ${a0}
lattice diamond 5.430949777364731
Lattice spacing in x,y,z = 5.43095 5.43095 5.43095
region           supercell block 0 10 -5 7 5 7 units lattice
create_box       1 supercell
Created orthogonal box = (0 -27.1547 27.1547) to (54.3095 38.0166 38.0166)
  1 by 1 by 1 MPI processor grid
region           simbox block 0 10 0 2 0 2 units lattice
region           nanowire block 0 10 0 2 0 2 units lattice
create_atoms     1 region nanowire
Created 0 atoms
  create_atoms CPU = 0.000153 secs
mass             1 28.0855
kim_interactions Si
#=== BEGIN kim_interactions ==============================
pair_style kim SW_StillingerWeber_1985_Si__MO_405512056662_005
pair_coeff * * Si
#=== END kim_interactions ================================


thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0
```

Using a KIM Model

- Using a KIM

```
kim_init
boundary
kim_query

lattice
region
create_box
region
region
create_ato
mass
kim_intera

thermo 10
thermo_sty
min_style
fix 1 all

minimize 1
```

```
LAMMPS (7 Aug 2019)
OMP_NUM_THREADS environment is not set. Defaulting to 1 thread. (src/comm.cpp:93)
  using 1 OpenMP thread(s) per MPI task
kim_init SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
#=== BEGIN kim-init ======================================
units metal
#=== END kim-init ========================================

boundary p p p
kim_query a0 get_lattice_constant_cubic crystal=["diamond"] species=["Si"] units=["angstrom"]
#=== BEGIN kim-query =====================================
variable a0 string 5.430949777364731
#=== END kim-query =======================================

lattice diamond ${a0}
lattice diamond 5.430949777364731
Lattice spacing in x,y,z = 5.43095 5.43095 5.43095
region           supercell block 0 10 -5 7 5 7 units lattice
create_box       1 supercell
Created orthogonal box = (0 -27.1547 27.1547) to (54.3095 38.0166 38.0166)
  1 by 1 by 1 MPI processor grid
region           simbox block 0 10 0 2 0 2 units lattice
region           nanowire block 0 10 0 2 0 2 units lattice
create_atoms     1 region nanowire
Created 0 atoms
  create_atoms CPU = 0.000153 secs
mass             1 28.0855
kim_interactions Si
#=== BEGIN kim_interactions ==============================
pair_style kim SW_StillingerWeber_1985_Si__MO_405512056662_005
pair_coeff * * Si
#=== END kim_interactions ================================


thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0
```

Using a KIM Model

- Using a KIM

```
kim_init
boundary
kim_query

lattice
region
create_box
region
region
create_ator
mass
kim_intera

thermo 10
thermo_sty
min_style
fix 1 all

minimize 1
```

```
LAMMPS (7 Aug 2019)
OMP_NUM_THREADS environment is not set. Defaulting to 1 thread. (src/comm.cpp:93)
  using 1 OpenMP thread(s) per MPI task
kim_init SW_StillingerWeber_1985_Si__MO_405512056662_005 metal
#=== BEGIN kim-init ======================================
units metal
#=== END kim-init ========================================

boundary p p p
kim_query a0 get_lattice_constant_cubic crystal=["diamond"] species=["Si"] units=["angstrom"]
#=== BEGIN kim-query =====================================
variable a0 string 5.430949777364731
#=== END kim-query =======================================

lattice diamond ${a0}
lattice diamond 5.430949777364731
Lattice spacing in x,y,z = 5.43095 5.43095 5.43095
region           supercell block 0 10 -5 7 5 7 units lattice
create_box       1 supercell
Created orthogonal box = (0 -27.1547 27.1547) to (54.3095 38.0166 38.0166)
  1 by 1 by 1 MPI processor grid
region           simbox block 0 10 0 2 0 2 units lattice
region           nanowire block 0 10 0 2 0 2 units lattice
create_atoms     1 region nanowire
Created 0 atoms
  create_atoms CPU = 0.000153 secs
mass             1 28.0855
kim_interactions Si
#=== BEGIN kim_interactions ==============================
pair_style kim SW_StillingerWeber_1985_Si__MO_405512056662_005
pair_coeff * * Si
#=== END kim_interactions ================================


thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
min_style cg
fix 1 all box/relax x 0.0 fixedpoint 0 0 0
```

Using a KIM Model

- Using a KIM

```
kim_init
boundary
kim_query

lattice
region
create_box
region
region
create_ator
mass
kim_intera

thermo 10
thermo_sty
min_style
fix 1 all

minimize 1
```

```
LAMMPS (7 Aug 2019)
OMP_NUM_THREADS environment is not
  using 1 OpenMP thread(s) per MPI
kim_init SW_StillingerWeber_1985_S
#=== BEGIN kim-init ==============
units metal
#=== END kim-init ================

boundary p p p
kim_query a0 get_lattice_constant_
#=== BEGIN kim-query =============
variable a0 string 5.4309497773647
#=== END kim-query ===============

lattice diamond ${a0}
lattice diamond 5.430949777364731
Lattice spacing in x,y,z = 5.43095
region           supercell block 0
create_box       1 supercell
Created orthogonal box = (0 -27.15
  1 by 1 by 1 MPI processor grid
region           simbox block 0 10
region           nanowire block 0
create_atoms     1 region nanowire
Created 0 atoms
  create_atoms CPU = 0.000153 secs
mass             1 28.0855
kim_interactions Si
#=== BEGIN kim_interactions ======
pair_style kim SW_StillingerWeber_
pair_coeff * * Si
#=== END kim_interactions ========

thermo 10
thermo_style custom step lx ly lz
min_style cg
fix 1 all box/relax x 0.0 fixedpoi
```

```
minimize 1.0e-16 1.0e-16 5000 10000
WARNING: Using 'neigh_modify every 1 delay 0 check yes' setting during minimization (src/min.cpp:168)
Neighbor list info ...
  update every 1 steps, delay 0 steps, check yes
  max neighbors/atom: 2000, page size: 100000
  master list distance cutoff = 5.77118
  ghost atom cutoff = 5.77118
  binsize = 2.88559, bins = 19 23 4
  1 neighbor lists, perpetual/occasional/extra = 1 0 0
  (1) pair kim, perpetual
      attributes: full, newton off, cut 5.77118
      pair build: full/bin/atomonly
      stencil: full/bin/3d
      bin: standard
Per MPI rank memory allocation (min/avg/max) = 0.4545 | 0.4545 | 0.4545 Mbytes
Step Lx Ly Lz Press Pxx Pyy Pzz PotEng
      0    54.309498    65.171397      10.8619            0            0            0            0            0
      1    54.309498    65.171397      10.8619            0            0            0            0            0
Loop time of 5e-05 on 1 procs for 1 steps with 0 atoms

36.0% CPU use with 1 MPI tasks x 1 OpenMP threads

Minimization stats:
  Stopping criterion = search direction is not downhill
  Energy initial, next-to-last, final =
                    0                    0                    0
  Force two-norm initial, final = 0 0
  Force max component initial, final = 0 0
  Final line search alpha, max atom move = 7.22762e+159 0
  Iterations, force evaluations = 1 0

MPI task timing breakdown:
Section |  min time  |  avg time  |  max time  |%varavg| %total
---------------------------------------------------------------
Pair    | 6e-06      | 6e-06      | 6e-06      |   0.0 | 12.00
Neigh   | 0          | 0          | 0          |   0.0 |  0.00
Comm    | 0          | 0          | 0          |   0.0 |  0.00
Output  | 0          | 0          | 0          |   0.0 |  0.00
Modify  | 0          | 0          | 0          |   0.0 |  0.00
Other   |            | 4.4e-05    |            |       | 88.00

Nlocal:    0 ave 0 max 0 min
Histogram: 1 0 0 0 0 0 0 0 0 0
Nghost:    0 ave 0 max 0 min
Histogram: 1 0 0 0 0 0 0 0 0 0
Neighs:    0 ave 0 max 0 min
Histogram: 1 0 0 0 0 0 0 0 0 0
FullNghs:  0 ave 0 max 0 min
Histogram: 1 0 0 0 0 0 0 0 0 0

Total # of neighbors = 0
Neighbor list builds = 0
Dangerous builds = 0

Please see the log.cite file for references relevant to this simulation

Total wall time: 0:00:00
```

log.cite

```
This LAMMPS simulation made specific use of work described in the
following references.  See http://lammps.sandia.gov/cite.html
for details.

@Comment
{
\documentclass{article}
\usepackage{url}
\begin{document}
This Model originally published in \cite{MO_405512056662_005a, MO_405512056662_005b} is archived in
OpenKIM~\cite{MO_405512056662_005, MD_335816936951_004, tadmor:elliott:2011, elliott:tadmor:2011}.
\bibliographystyle{vancouver}
\bibliography{kimcite-MO_405512056662_005.bib}
\end{document}
}

@Misc{MO_405512056662_005,
  author       = {Amit K Singh},
  title        = {{S}tillinger-{W}eber potential for {S}i due to {S}tillinger and {W}eber (1985) v005},
  doi          = {10.25950/c74b293f},
  howpublished = {OpenKIM, \url{https://doi.org/10.25950/c74b293f}},
  keywords     = {OpenKIM, Model, MO_405512056662_005},
  publisher    = {OpenKIM},
  year         = 2018,
}

@Misc{MD_335816936951_004,
  author       = {Mingjian Wen},
  title        = {{S}tillinger-{W}eber ({S}{W}) {M}odel {D}riv
  doi          = {10.25950/f3abd2d6},
  howpublished = {OpenKIM, \url{https://doi.org/10.25950/c74b2
  keywords     = {OpenKIM, Model Driver, MD_335816936951_004},
  publisher    = {OpenKIM},
  year         = 2018,
}

@Article{tadmor:elliott:2011,
  author    = {E. B. Tadmor and R. S. Elliott and J. P. Sethna
  title     = {The potential of atomistic simulations and the
  journal   = {{JOM}},
  year      = {2011},
  volume    = {63},
  number    = {7},
  pages     = {17},
  doi       = {10.1007/s11837-011-0102-6},
}
```

```
@Misc{elliott:tadmor:2011,
  author       = {Ryan S. Elliott and Ellad B. Tadmor},
  title        = {{K}nowledgebase of {I}nteratomic {M}odels ({KIM}) Application Programming Interface ({API})},
  howpublished = {\url{https://openkim.org/kim-api}},
  publisher    = {OpenKIM},
  year         = 2011,
  doi          = {10.25950/ff8f563a},
}

@Article{MO_405512056662_005a,
  author = {Stillinger, Frank H. and Weber, Thomas A.},
  doi = {10.1103/PhysRevB.31.5262},
  issue = {8},
  journal = {Physical Review B},
  month = {Apr},
  pages = {5262--5271},
  publisher = {American Physical Society},
  title = {Computer simulation of local order in condensed phases of silicon},
  volume = {31},
  year = {1985},
}

@Book{MO_405512056662_005b,
  author = {Tadmor, Ellad B. and Miller, Ronald E.},
  doi = {10.1017/CBO9781139003582},
  publisher = {Cambridge University Press},
  title = {Modeling Materials: {C}ontinuum, Atomistic and Multiscale Techniques},
  year = {2011},
}
```

log.cite

```
This LAMMPS simulation made specific use of work described in the
following references.  See http://lammps.sandia.gov/cite.html
for details.

@Comment
{
\documentclass{
\usepackage{url
\begin{document
This Model orig
OpenKIM~\cite{M
\bibliographyst
\bibliography{k
\end{document}
}

@Misc{MO_405512
  author
  title
  doi
  howpublished
  keywords
  publisher
  year
}

@Misc{MD_335816
  author
  title
  doi
  howpublished
  keywords
  publisher
  year
}

@Article{tadmor
  author    = {
  title     = {
  journal   = {
  year      = {
  volume    = {
  number    = {
  pages     = {
  doi       = {
}
```

OpenKIM · SW StillingerWeber

https://openkim.org/id/SW_StillingerWeber_1985_Si__MO_405512056662_005

Search

SW_StillingerWeber_1985_Si__MO_405512056662_005

Title	Stillinger-Weber potential for Si due to Stillinger and Weber (1985) v005
Description	This is the original Stillinger-Weber parameterization for Si due to Stillinger and Weber (1985). The cohesive energy diamond Si obtained from this model is 4.3364 eV. This Model corresponds to the Si.sw parameter file distributed with the LAMMPS package except that Si.sw uses epsilon=2.1683 and the KIM model uses 2.1682 (converted more precisely from 50 kcal/mol given in the SW paper). However, given the low accuracy of the source data, either is acceptable. Due to this difference, the Si.sw file from LAMMPS and the KIM model give slightly different results scaled by 2.1683/2.1682.
Species	Si
Disclaimer	None
Contributor	amit
Maintainer	amit
Author	Amit K Singh
Publication Year	2018
Item Citation	This Model originally published in [1-2] is archived in OpenKIM [3-6]. [1] Stillinger FH, Weber TA. Computer simulation of local order in condensed phases of silicon. Physical Review B. 1985Apr;31(8):5262–71. doi:10.1103/PhysRevB.31.5262 [2] Tadmor EB, Miller RE. Modeling Materials: Continuum, Atomistic and Multiscale Techniques. Cambridge University Press; 2011. doi:10.1017/CBO9781139003582 [3] Singh AK. Stillinger-Weber potential for Si due to Stillinger and Weber (1985) v005. OpenKIM; 2018. doi:10.25950/c74b293f [4] Wen M. Stillinger-Weber (SW) Model Driver v004. OpenKIM; 2018. doi:10.25950/f3abd2d6 [5] 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 [6] Elliott RS, Tadmor EB. Knowledgebase of Interatomic Models (KIM) Application Programming Interface (API). OpenKIM; 2011. doi:10.25950/ff8f563a Click here to download the above citation in BibTeX format.

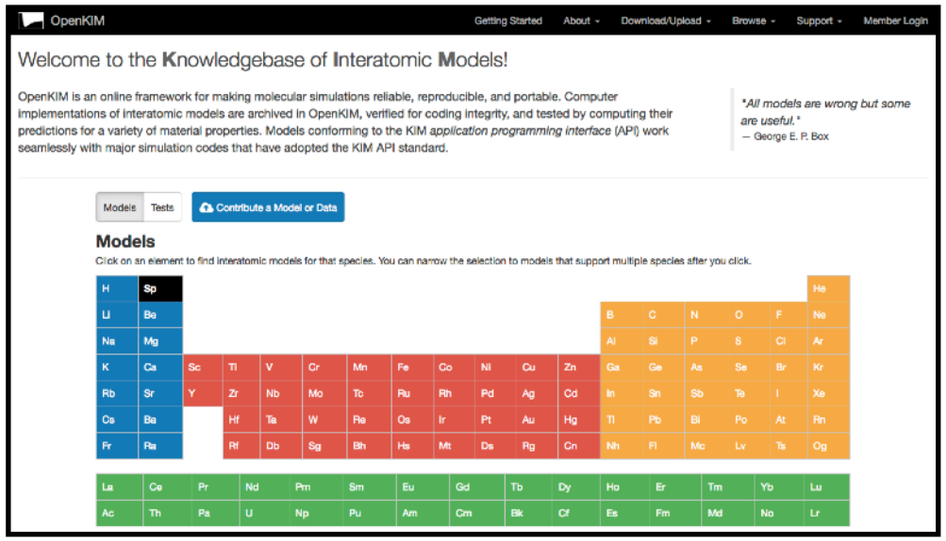
```
face ({API})},
```

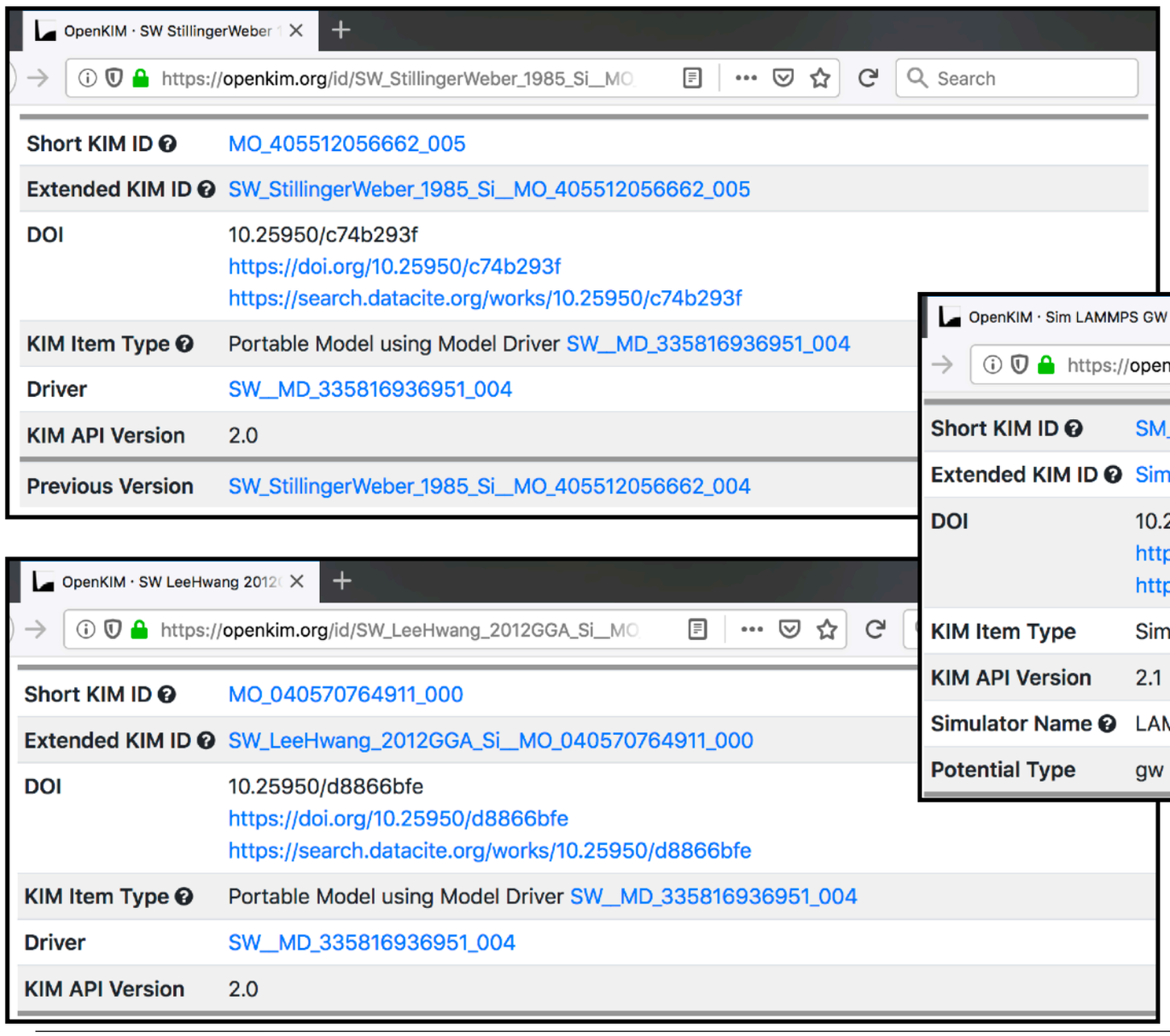
```
  title = {Modeling Materials: {C}ontinuum, Atomistic and Multiscale Techniques},
  year = {2011},
}
```

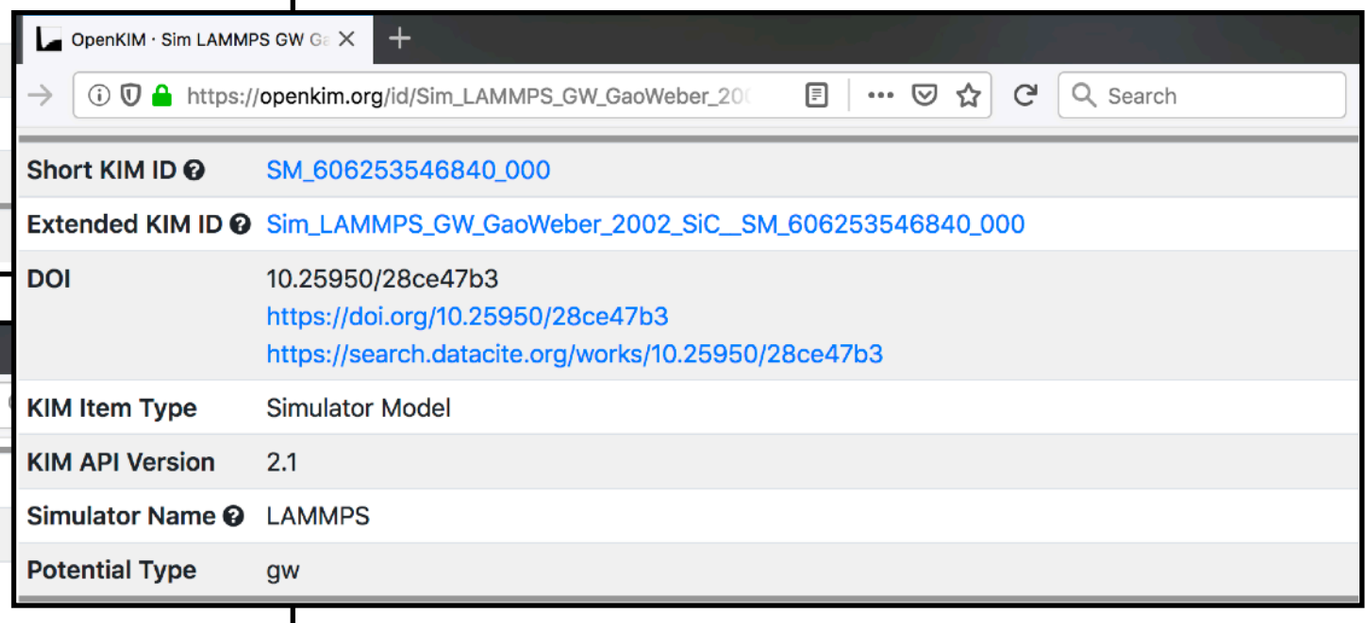
How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

Compare Models using KIM







Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"      &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"          &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to units specied in kim_init

# Set variables, construct lattice, create imperfection, ...
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Par): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration*
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

* Dump file coordinate units are not converted.

Compare Models using KIM

```
# Silicon nanowire --- static uniaxial tension along x-direction

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512
                     "SW_LeeHwang_2012GGA_Si__MO_0405707649
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_6

label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
...

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond
                                           units ["angstrom
variable a0 equal ${a0_A}*${_u_distance}  # Convert to unit

# Set variables, construct lattice, create imperfection, ..
...

# Specify atom type mapping for KIM model
kim_interactions Si

...

# Define looping variables
variable n loop ${number_load_steps}
label loop
  # Compute stretch factor, total strain, stretch box, ...
  ...

  # Minimize positions according to zero stress along x dir
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_
  print "STEP ${n}; strain: ${total_strain}; engineering st
  print "${total_strain} ${pressf}" append out_stress-strai

  # Dump relaxed strained configration*
  ...
next n
jump SELF loop
print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop
print "All simulations complete"
```

```
...

kim_init ${MODEL} metal unit_conversion_mode
kim_init Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000 metal unit_conversion_mode
#=== BEGIN kim-init ======================================
# Using KIM Simulator Model : Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000
# For Simulator             : LAMMPS 22 Sep 2017
# Running on                : LAMMPS 7 Aug 2019
#
# Conversion factors from metal to metal:
variable _u_mass         internal 1.000000000000e+00
variable _u_distance     internal 1.000000000000e+00
variable _u_time         internal 1.000000000000e+00
variable _u_energy       internal 1.000000000000e+00
variable _u_velocity     internal 1.000000000000e+00
variable _u_force        internal 1.000000000000e+00
variable _u_torque       internal 1.000000000000e+00
variable _u_temperature  internal 1.000000000000e+00
variable _u_pressure     internal 1.000000000000e+00
variable _u_viscosity    internal 1.000000000000e+00
variable _u_charge       internal 1.000000000000e+00
variable _u_dipole       internal 1.000000000000e+00
variable _u_efield       internal 1.000000000000e+00
variable _u_density      internal 1.000000000000e+00
#
units metal
#=== END kim-init ========================================

dimension 3
boundary p p p
atom_style atomic

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] species=["Si"] units=["angstrom"]
#=== BEGIN kim-query ======================================
variable a0_A string 5.224877543747425
#=== END kim-query ========================================

variable a0 equal ${a0_A}*${_u_distance}  # Convert to unit system specied in kim_init line
variable a0 equal 5.224877543747425*${_u_distance}
variable a0 equal 5.224877543747425*1
variable wire_len_x equal 10              # Nanowire length in x-direction (in unit cells)
variable wire_len_y equal 2               # Nanowire length in y-direction (in unit cells)
variable wire_len_z equal 2               # Nanowire length in z-direction (in unit cells)
variable strain_increment equal 0.01      # Increment of strain relative to ref config
                                          # at each load step
variable number_load_steps equal 200      # Number of load steps

...

# Set mass to dummy value (not used in static minimization)
mass 1 1.0

# Specify atom type mapping for KIM model
kim_interactions Si
#=== BEGIN kim_interactions ===============================
pair_style gw
pair_coeff * * /var/tmp/kim-simulator-model-parameter-file-directory-Zoja9KAfgfeS/SiC.gw Si
Reading potential file /var/tmp/kim-simulator-model-parameter-file-directory-Zoja9KAfgfeS/SiC.gw with DATE: 2016-05-06
#=== END kim_interactions =================================

...
```

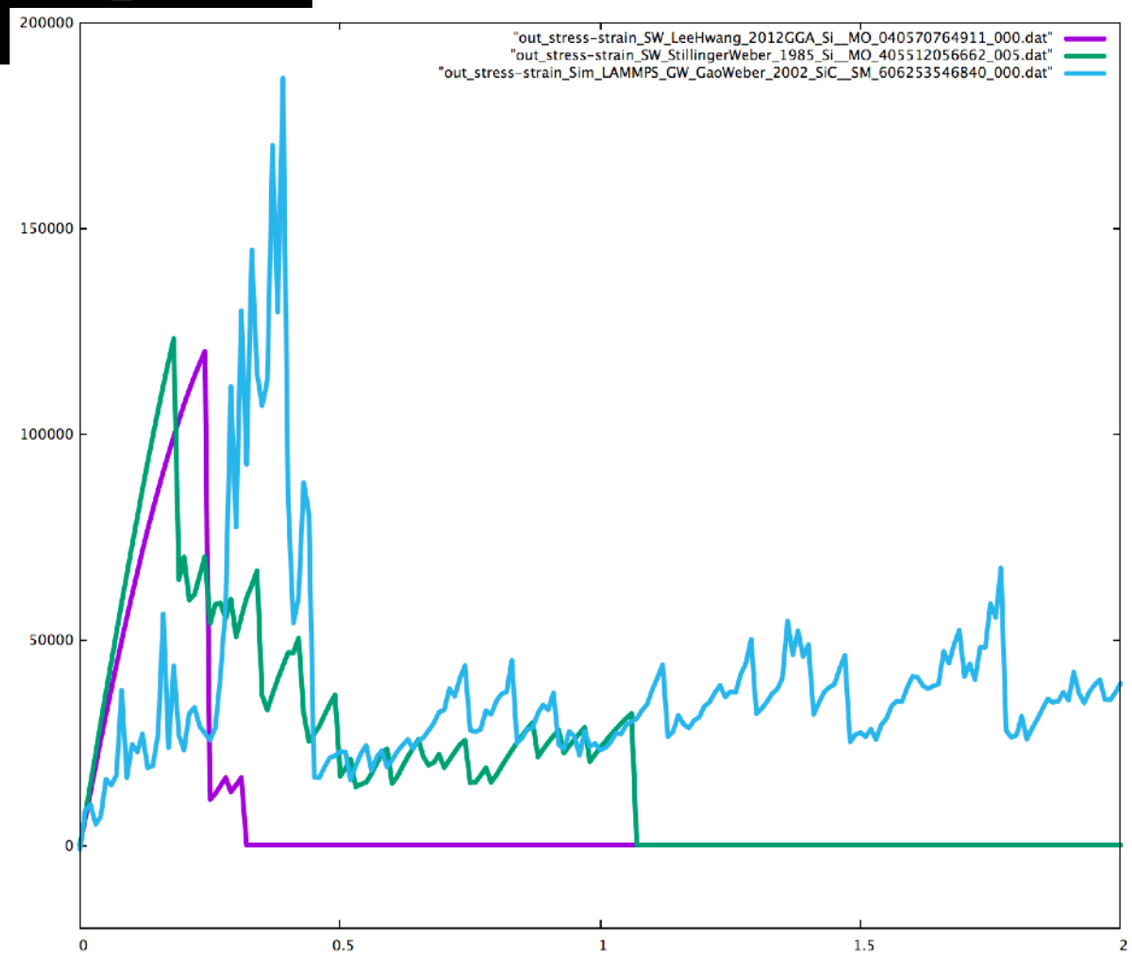
✳ Dump file coordinate units are not converted.

Complete script

```
# Silicon nanowire --- static uniaxial tension along x-direction
#
# Based on the example by Mark Tschopp and Nathan Rhodes available at:
#   https://icme.hpc.msstate.edu/mediawiki/index.php/LAMMPS_Fracture
#
# Authors: Daniel S. Karls, Ellad B. Tadmor, Ryan S. Elliott
# Last revision: 8/8/2019

variable MODEL index "SW_StillingerWeber_1985_Si__MO_405512056662_005"       &
                     "SW_LeeHwang_2012GGA_Si__MO_040570764911_000"           &
                     "Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000"
label main_loop
clear

kim_init ${MODEL} metal unit_conversion_mode
dimension 3
boundary p p p
atom_style atomic

# Simulation variables
kim_query a0_A get_lattice_constant_cubic crystal=["diamond"] &
                                          species=["Si"]      &
                                          units=["angstrom"]
variable a0 equal ${a0_A}*${_u_distance}  # Convert to unit system specied in kim_init line
variable wire_len_x equal 10              # Nanowire length in x-direction (in unit cells)
variable wire_len_y equal 2               # Nanowire length in y-direction (in unit cells)
variable wire_len_z equal 2               # Nanowire length in z-direction (in unit cells)
variable strain_increment equal 0.01      # Increment of strain relative to ref config
                                          # at each load step
variable number_load_steps equal 200      # Number of load steps

# Derived/other variables
variable vacuum_pad equal 5               # Set the vacuum along the y and z directions
variable box_min_y equal -${vacuum_pad}
variable box_max_y equal ${wire_len_y}+${vacuum_pad}
variable box_min_z equal -${vacuum_pad}
variable box_max_z equal ${wire_len_z}+${vacuum_pad}
variable box_yz_area equal (${box_max_y}-${box_min_y})*(${box_max_z}-${box_min_z})*${a0}^2
variable wire_yz_area equal ${wire_len_y}*${wire_len_z}*${a0}^2

# Construct lattice
lattice diamond ${a0}
region supercell block 0.0          ${wire_len_x} &
                       ${box_min_y} ${box_max_y}  &
                       ${box_min_z} ${box_max_z} units lattice
create_box 1 supercell
region nanowire block EDGE EDGE 0.0 ${wire_len_y} 0.0 ${wire_len_z} units lattice
create_atoms 1 region nanowire

# Create imperfection
variable atomx   equal floor(${wire_len_x}/2)
variable atomy   equal 0.0
variable atomz   equal 0.0
variable atomrad equal 0.1
region void sphere ${atomx} ${atomy} ${atomz} ${atomrad}
delete_atoms region void

# Set mass to dummy value (not used in static minimization)
mass 1 1.0

# Specify atom type mapping for KIM model
kim_interactions Si

# First, perform a static minimization with respect to all atomic positions
# under the condition that the box size relax along the axial (x) direction in
# order to achieve zero stress along this dimension.
thermo 10
thermo_style custom step lx ly lz press pxx pyy pzz pe
fix 1 all box/relax x 0.0 fixedpoint 0 0 0

minimize 1.0e-16 1.0e-16 5000 10000

unfix 1


# Record stress
variable total_strain equal 0.0
variable pressf1 equal -pxx
variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
print "STEP 0; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
print "${total_strain} ${pressf}" file out_stress-strain_${MODEL}.dat

# Dump the relaxed refenence configuration prior to straining
dump dmp1 all atom 1 out_config_${MODEL}.dump
dump_modify dmp1 scale no
run 0
undump dmp1

############################################################
# Now execute an incremental strain along the axial dimension
############################################################

# Define looping variables
variable n loop ${number_load_steps}
label loop

  # Compute stretch factor relative to the deformed length needed to
  # obtain a strain increment of `strain_increment` relative to the
  # reference configuration (i.e. engineering or Lagrangian strain)
  variable stretch_factor equal &
           1.0+${strain_increment}/(1+${n}*${strain_increment})

  # Calculate total strain for this step
  variable total_strain equal ${n}*${strain_increment}

  # Stretch box
  change_box all x scale ${stretch_factor} remap

  # Minimize positions according to zero stress along x direction
  minimize 1.0e-10 1.0e-10 1000 10000

  # Print out stress
  variable pressf1 equal -pxx
  variable pressf equal ${pressf1}*${box_yz_area}/${wire_yz_area}/${_u_pressure}
  print "STEP ${n}; strain: ${total_strain}; engineering stress (Bar): ${pressf}"
  print "${total_strain} ${pressf}" append out_stress-strain_${MODEL}.dat

  # Dump relaxed strained configration
  dump dmp1 all atom 1 out_config_${MODEL}.dump
  dump_modify dmp1 append yes scale no
  run 0
  undump dmp1

next n
jump SELF loop

print "Simulation for ${MODEL} complete"

next MODEL
jump SELF main_loop

print "All simulations complete"
```

Run the script & compare

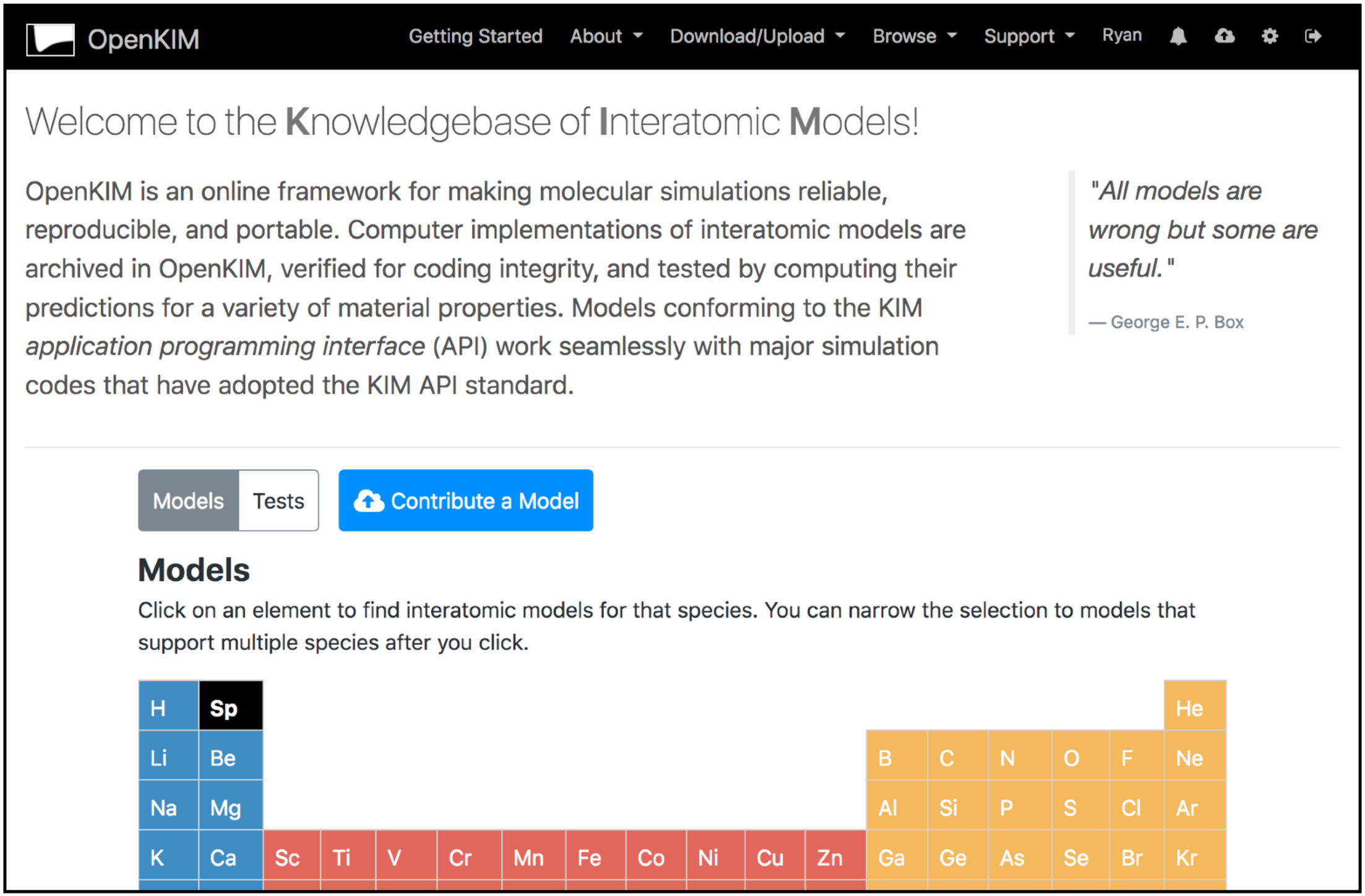
```
% ./lmp -in in-kim-si-nanowire-comparison > term.lammps 2>&1

% ls
in-kim-si-nanowire-comparison
kim.log
log.cite
log.lammps
out_config_SW_LeeHwang_2012GGA_Si__MO_040570764911_000.dump
out_config_SW_StillingerWeber_1985_Si__MO_405512056662_005.dump
out_config_Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000.dump
out_stress-strain_SW_LeeHwang_2012GGA_Si__MO_040570764911_000.dat
out_stress-strain_SW_StillingerWeber_1985_Si__MO_405512056662_005.dat
out_stress-strain_Sim_LAMMPS_GW_GaoWeber_2002_SiC__SM_606253546840_000.dat
term.lammps
```



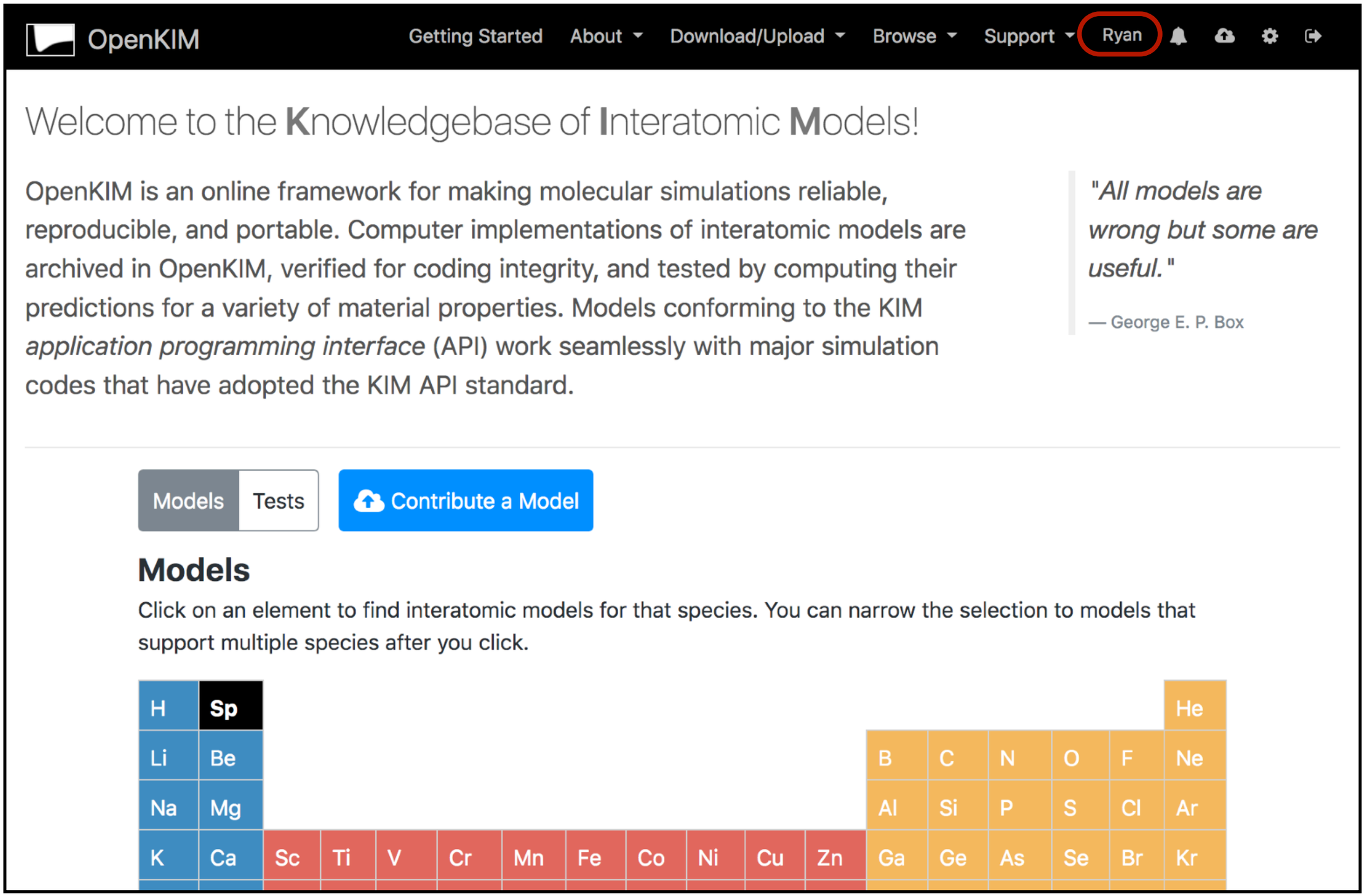
How To: Steps

1. Obtain LAMMPS (with KIM package) & OpenKIM Models
   a. Binaries
   b. Source (`openkim-models` or one-at-a-time)
2. Write LAMMPS scripts to perform your simulation
   - `kim_init`, `kim_query`, & `kim_interactions` commands
   - Modify your script to work with multiple models
3. Upload your Model to https://openkim.org [optional]

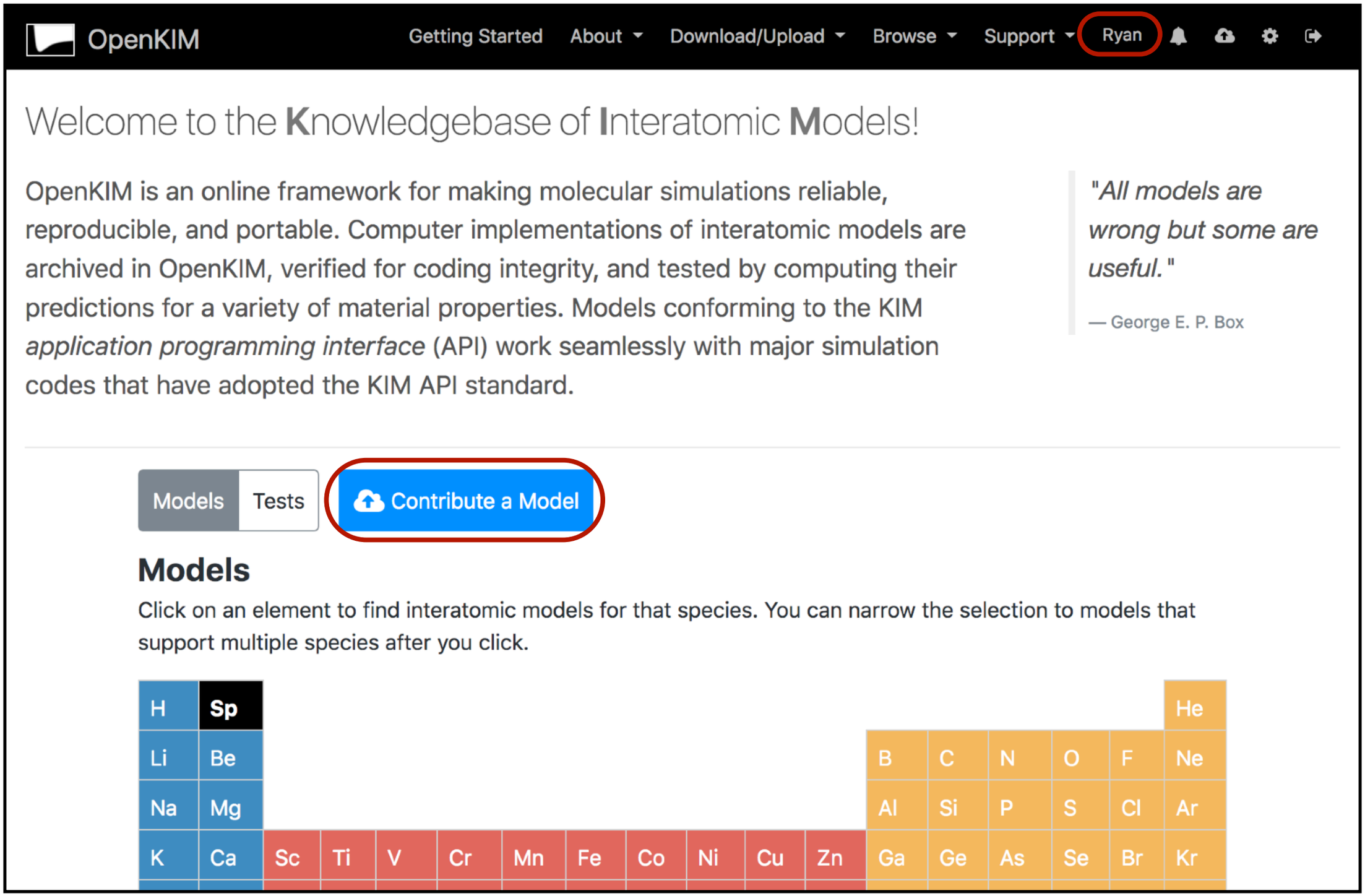
Upload a LAMMPS Simulator Model



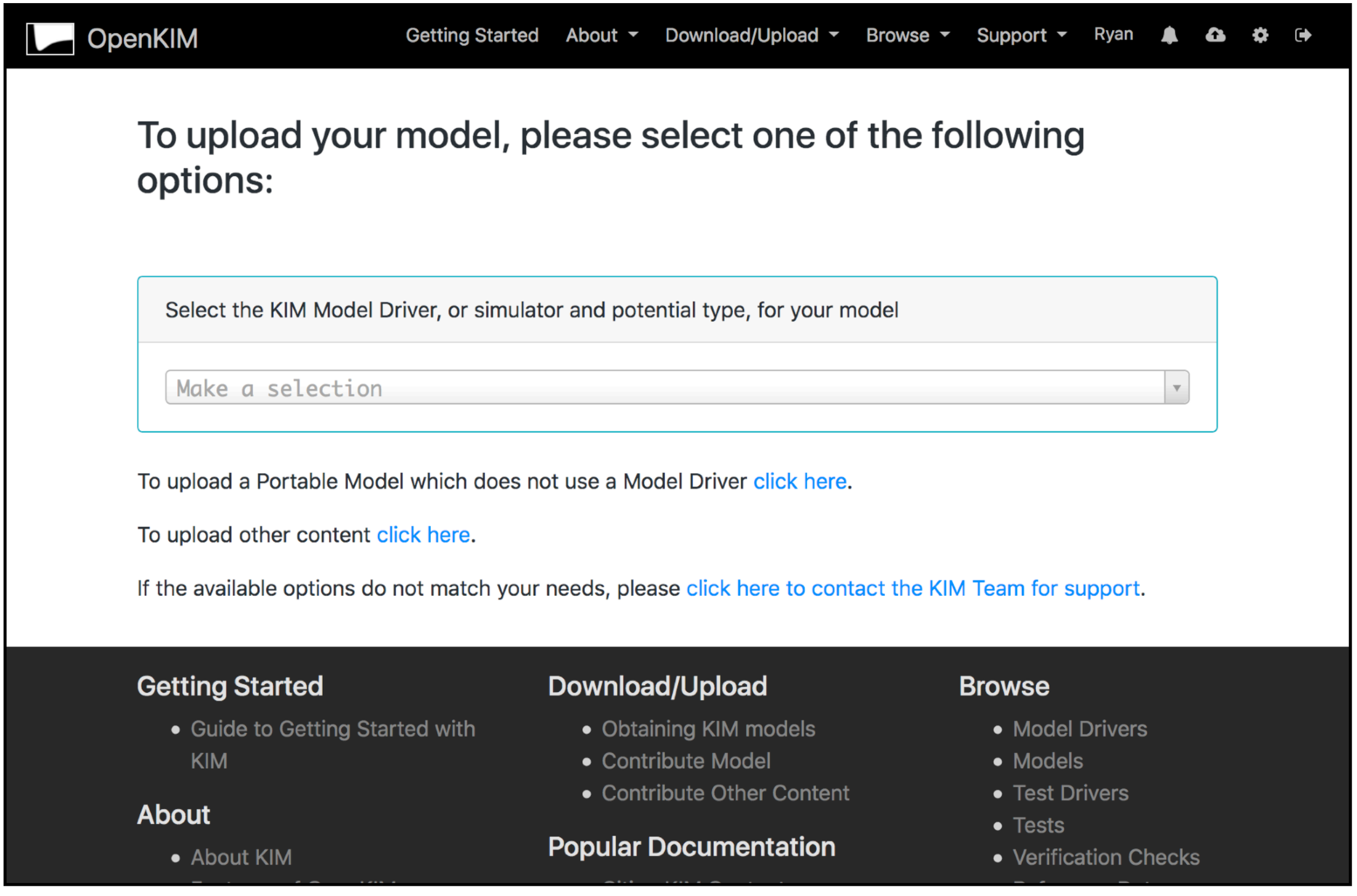
Upload a LAMMPS Simulator Model



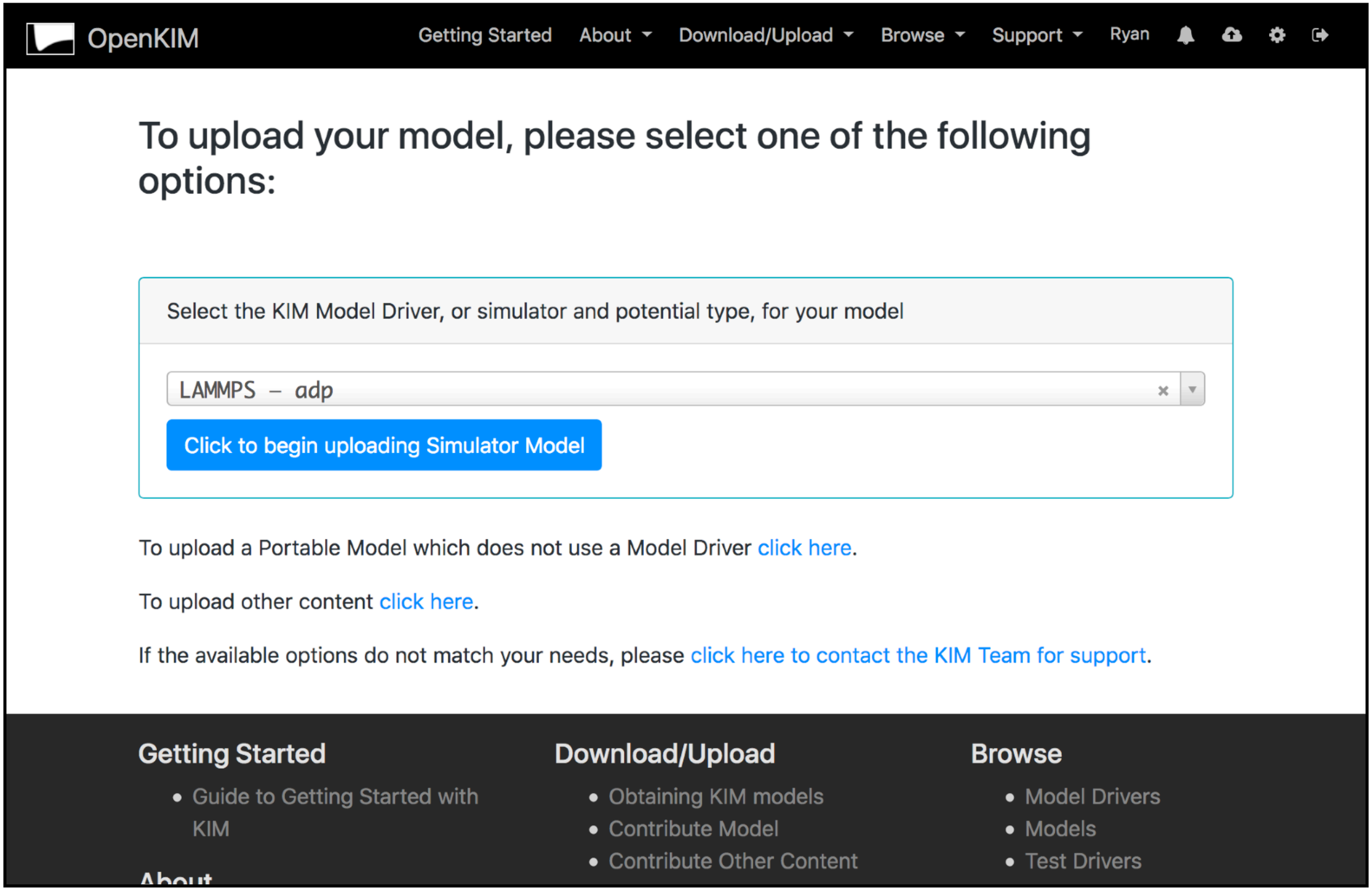
Upload a LAMMPS Simulator Model



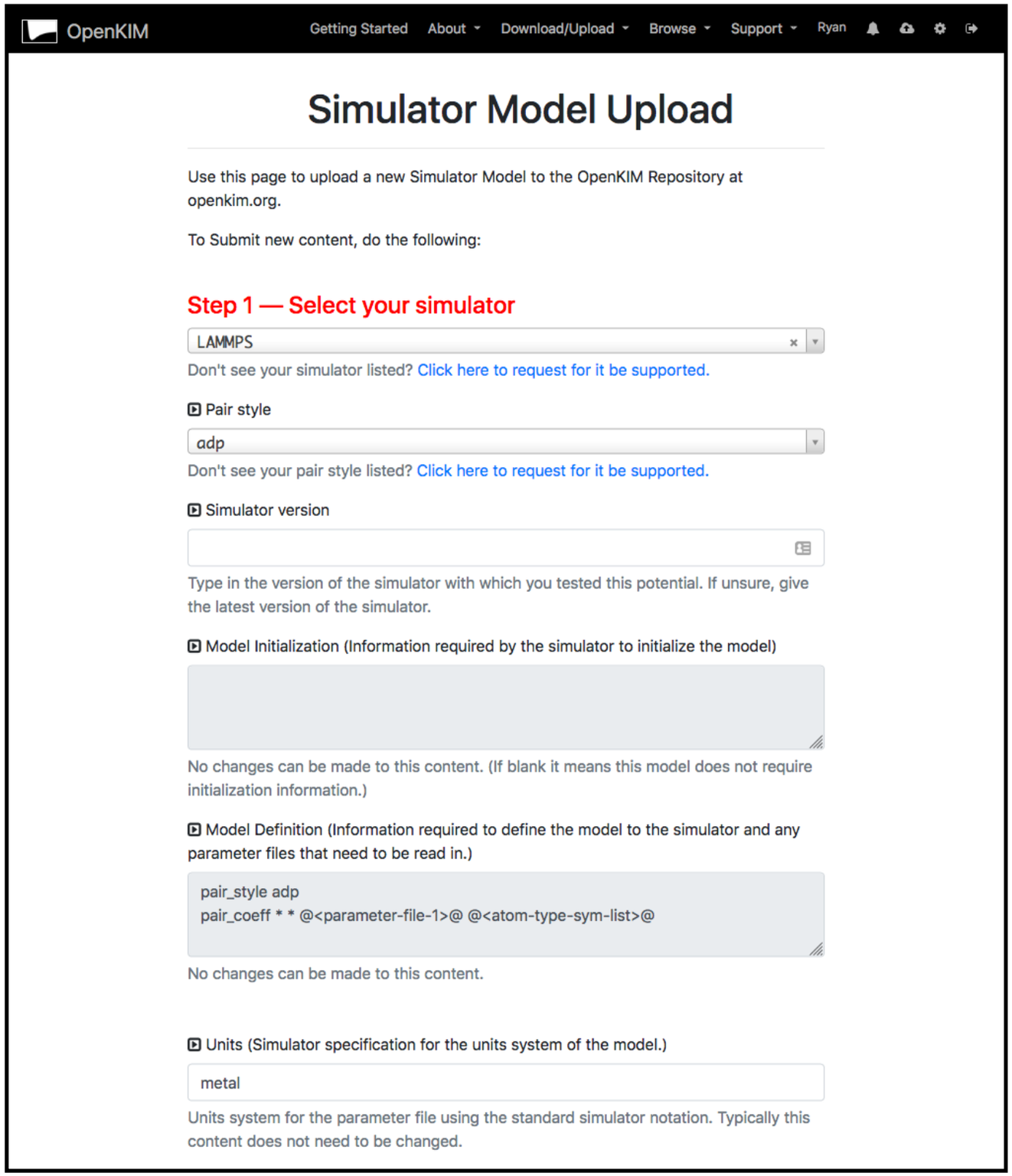
Upload a LAMMPS Simulator Model



Upload a LAMMPS Simulator Model

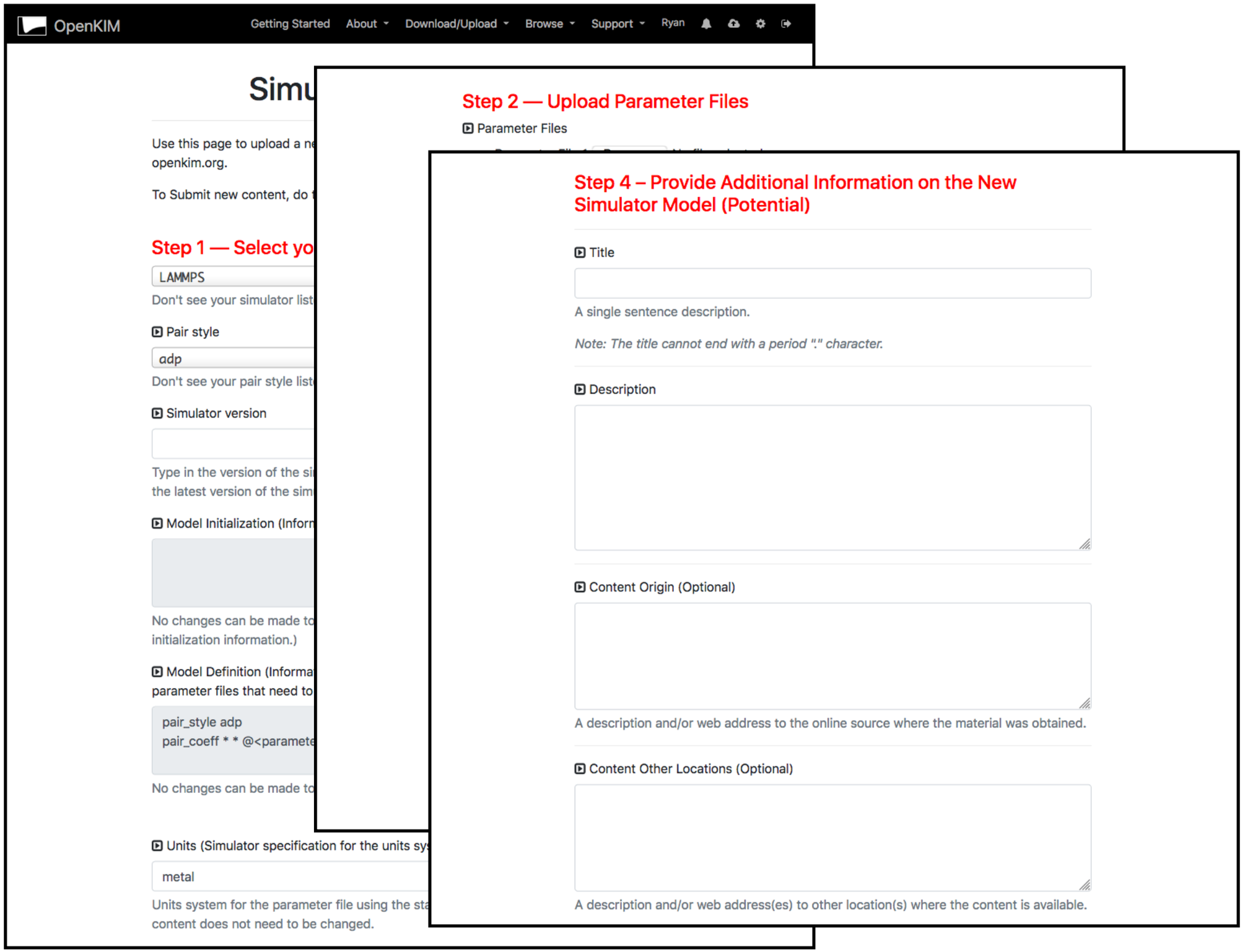


Upload a LAMMPS Simulator Model

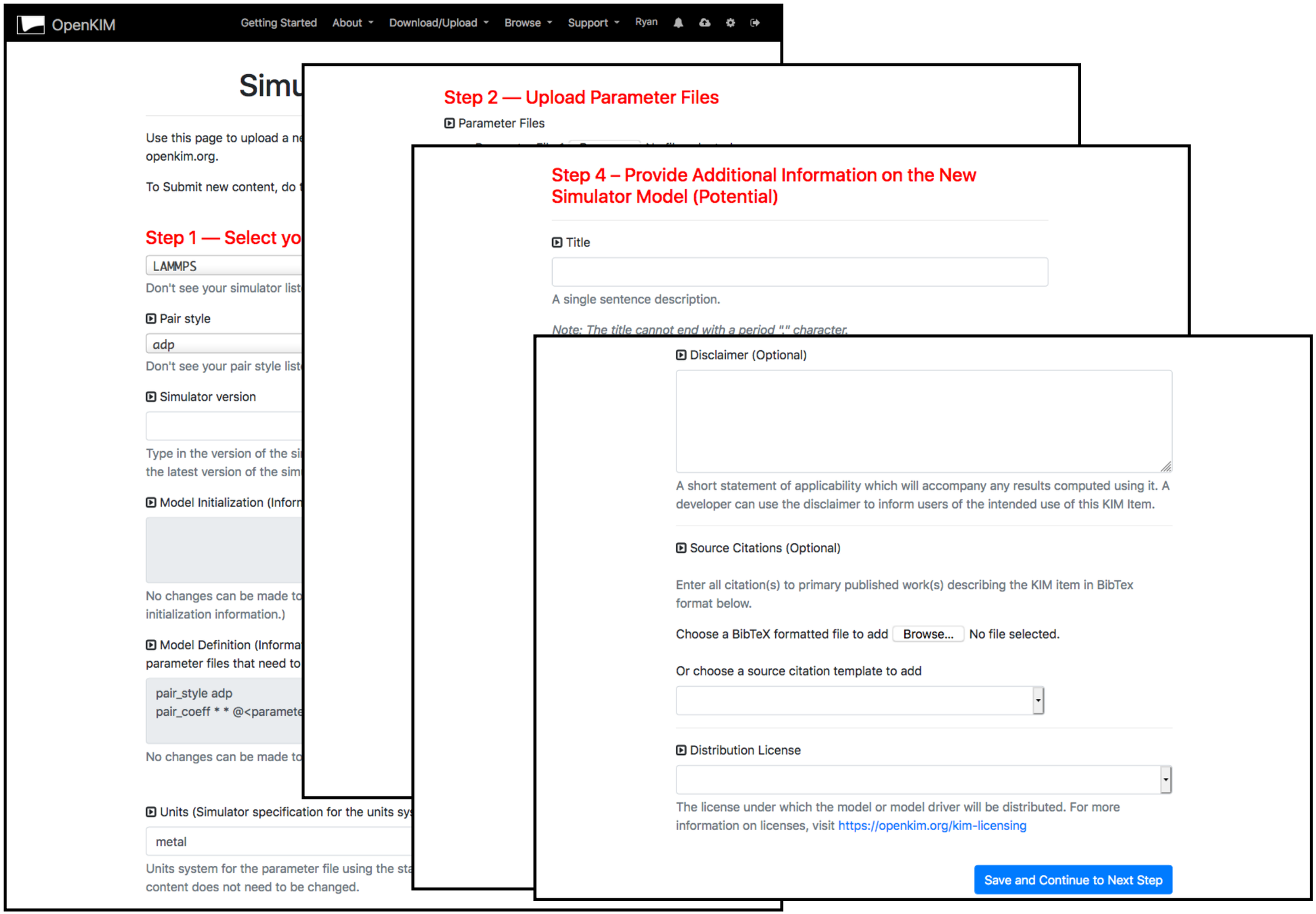


Upload a LAMMPS Simulator Model

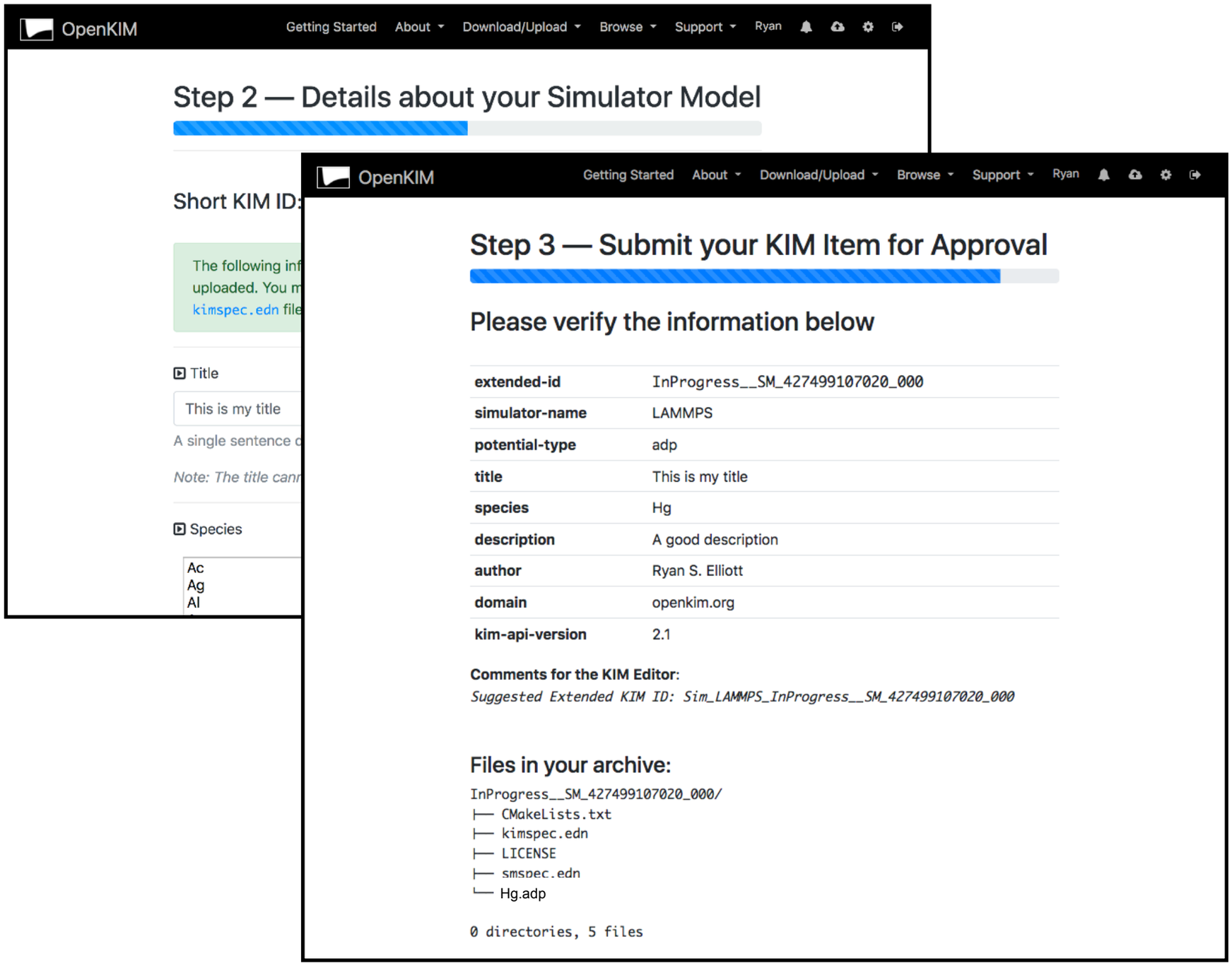
Upload a LAMMPS Simulator Model



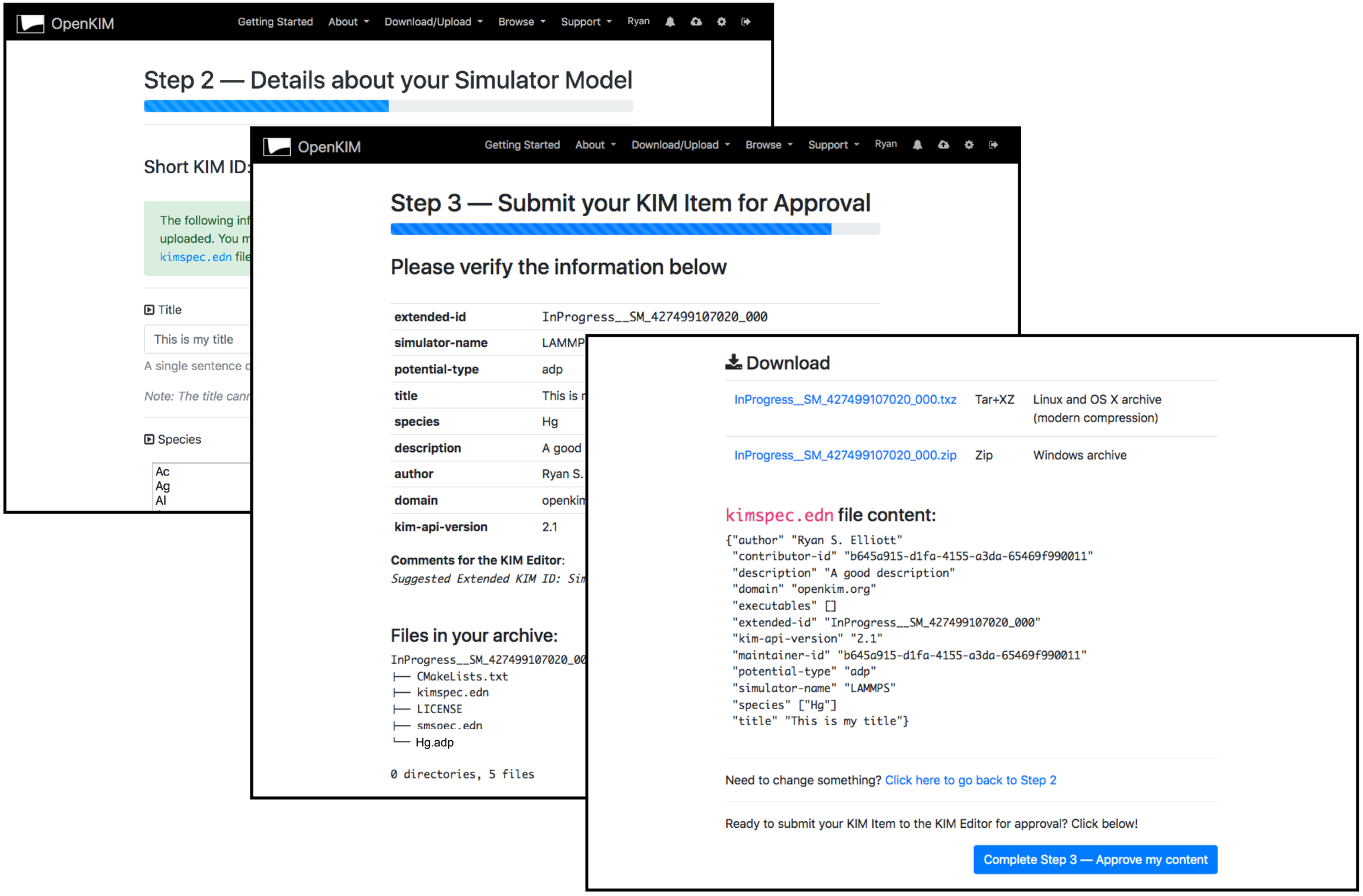
Upload a LAMMPS Simulator Model



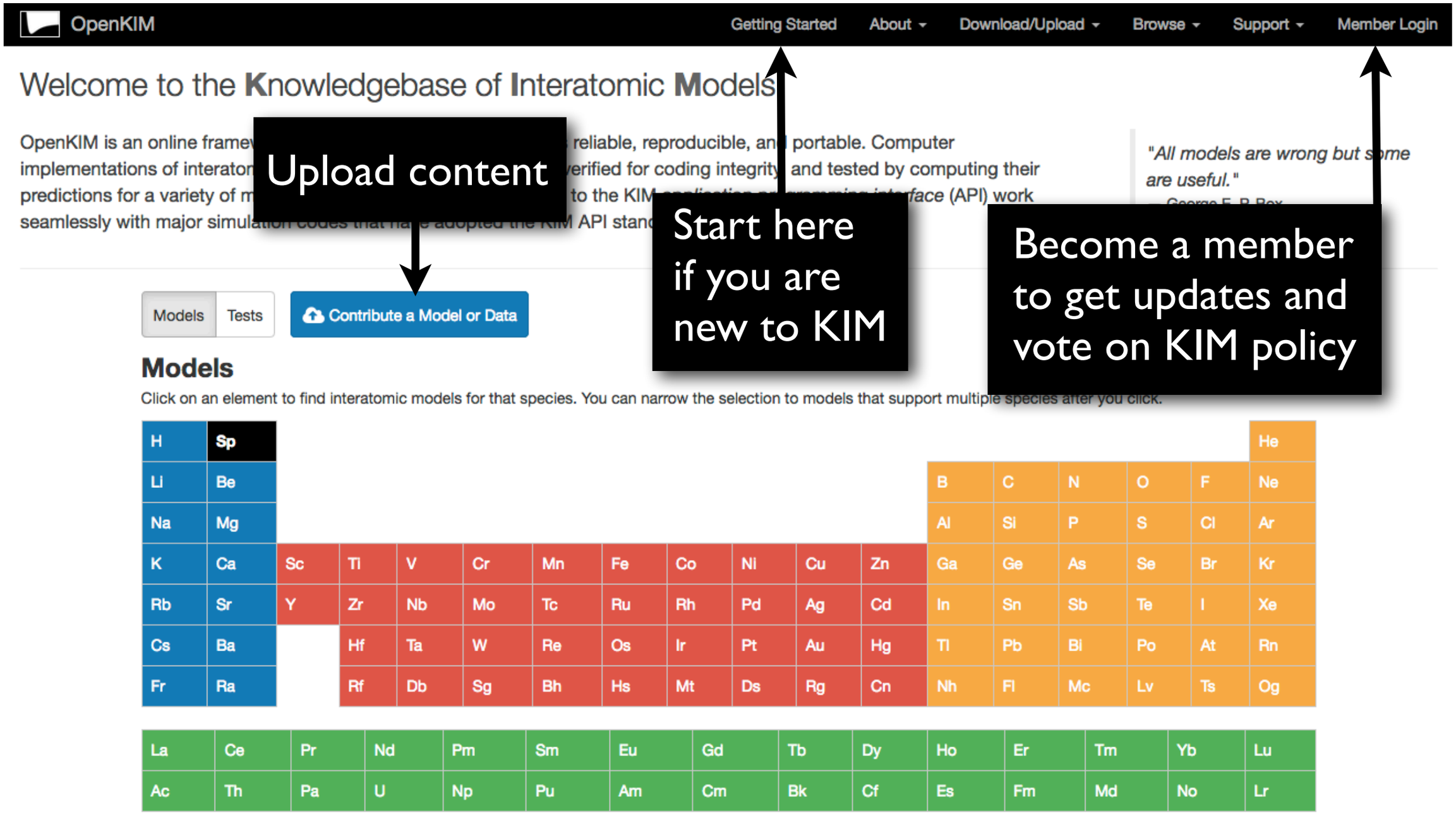
Upload a LAMMPS Simulator Model



Upload a LAMMPS Simulator Model



OpenKIM.org



https://openkim.org