The Knowledgebase of Interatomic Models (OpenKIM)

&

The Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)

Ryan S. Elliott

Department of Aerospace Engineering & Mechanics

The University of Minnesota

Ellad B. Tadmor (U. Minn), James P. Sethna (Cornell U.)

OUTLINE

1. The Knowledgebase of Interatomic Models
   - User Signup
   - KIM API
   - "KIM Compliant" Simulators
   - Setup VirtualBox Ubuntu VM
   - KIM Model performance within LAMMPS
   - KIM Model pages
2. Installing the KIM API on Ubuntu (ppa:openkim/openkim)
3. Using LAMMPS with the KIM API

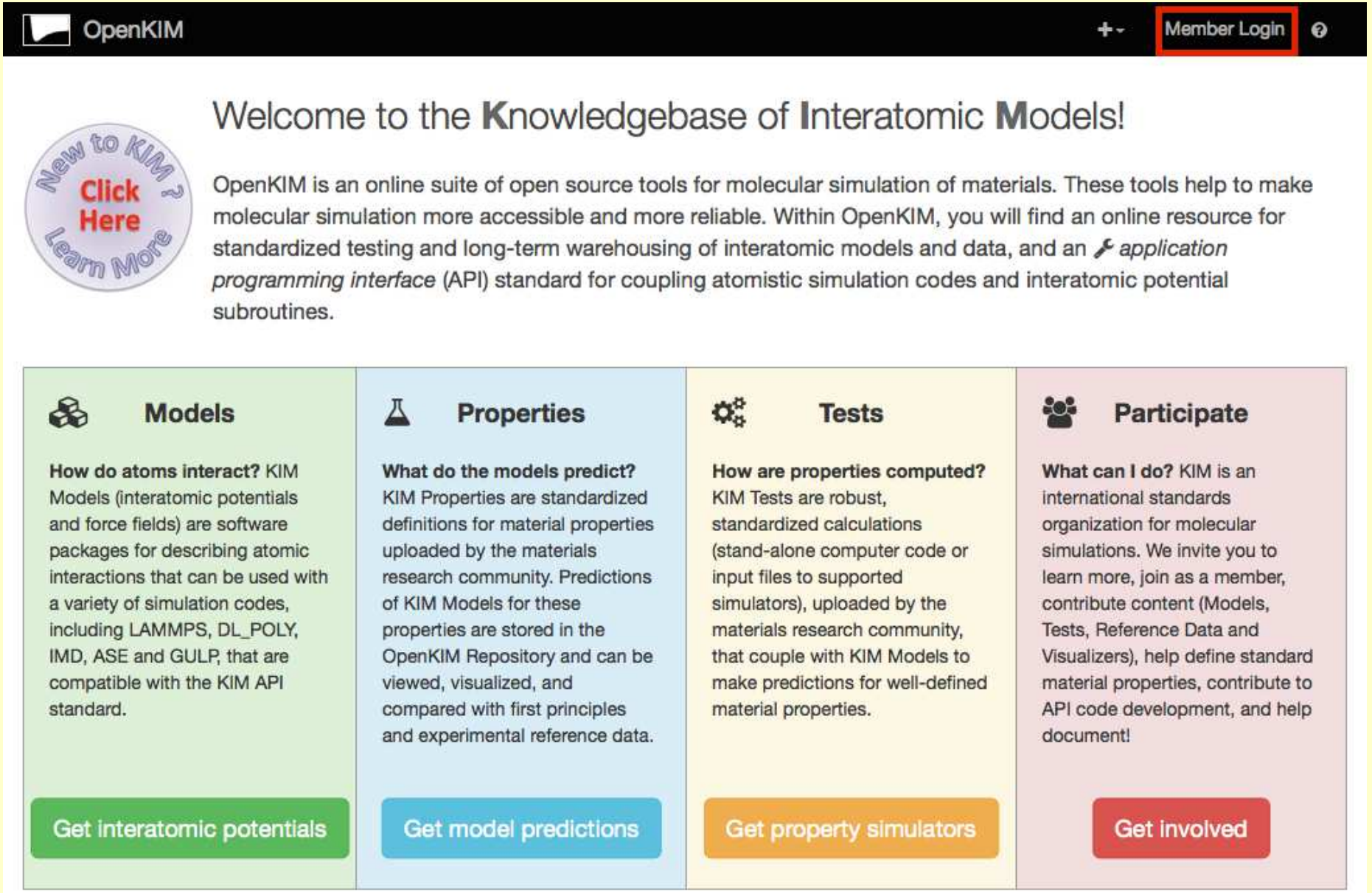
Appendix: Installing the KIM API by hand

OUTLINE

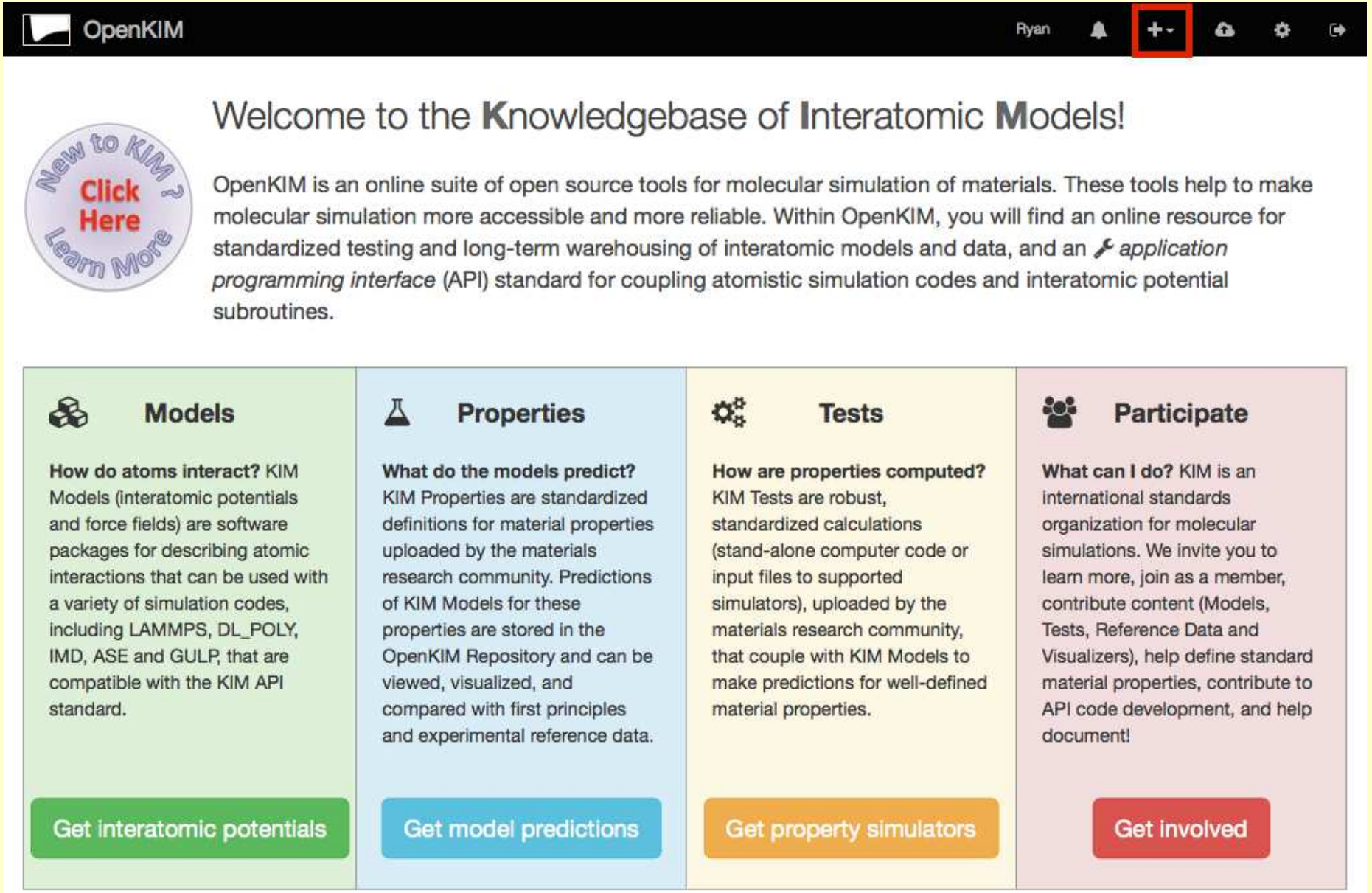
1. The Knowledgebase of Interatomic Models
   - User Signup
   - KIM API
   - "KIM Compliant" Simulators
   - Setup VirtualBox Ubuntu VM
   - KIM Model performance within LAMMPS
   - KIM Model pages
2. Installing the KIM API on Ubuntu (ppa:openkim/openkim)
3. Using LAMMPS with the KIM API

Appendix: Installing the KIM API by hand

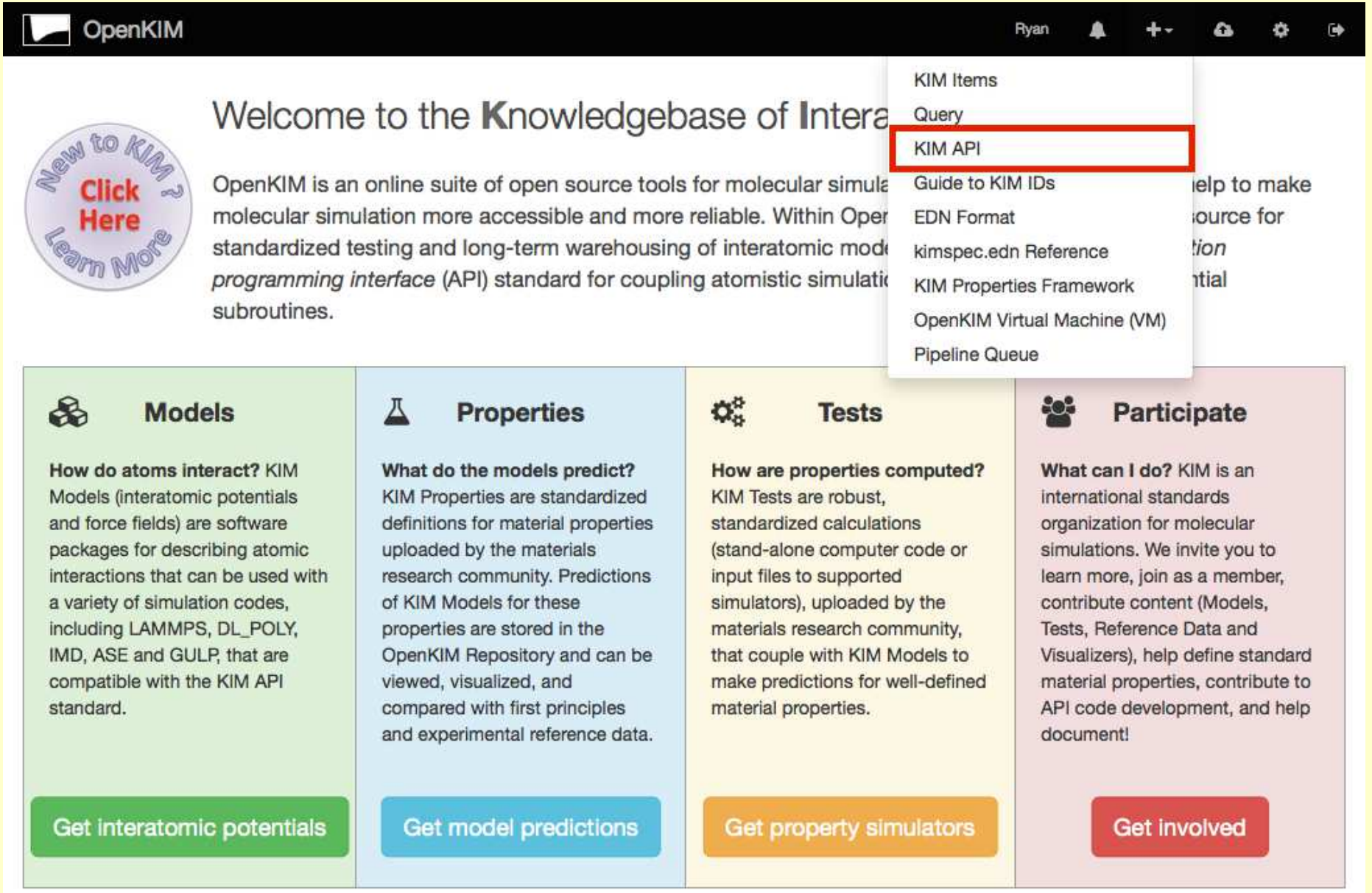
OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS

OpenKIM Ryan

KIM Application Programming Interface (API)

Overview

The KIM API is an Application Programming Interface for atomistic simulations. The API provides a standard for exchanging information between atomistic simulation codes (molecular dynamics, molecular statics, lattice dynamics, Monte Carlo, etc.) and interatomic models (potentials or force fields). It also includes a set of library routines for using the API with bindings for

- FORTRAN 77
- Fortran 90/95
- Fortran 2003
- C
- C++

By conforming to this API, an atomistic simulation code will seamlessly work with any KIM-compliant interatomic model written in any supported language. The interface is computationally efficient and often requires relatively minor changes to existing codes.

NOTE: It has been determined that **GCC 4.6.x** (and in particular, gfortran 4.6.x) has a number of insidious bugs associated with the use of Cray pointers *AND* the iso_c_binding interface. These bugs cause KIM Models to produce incorrect numerical results without generating any warnings or crashes. **Thus, gfortran 4.6.x SHOULD NOT BE USED with the openkim-api software package.**

Downloading the Most Recent API Package

The most recent packaged download of the KIM API is available here:

kim-api-latest

OpenKIM with LAMMPS

http://groups.google.com/group/openkim

Members of the OpenKIM development team actively monitor this forum and will do their best to respond to questions in a timely fashion. This forum is also used to announce minor new releases and bug fixes.

It is highly recommended for users who plan to work with the KIM API to become members of the openkim group. (Just go to the above link and click on "Join this group" on the right of the screen.)

About

- About KIM
- Getting Started with KIM
- What People Are Saying
- Funding
- Citing the KIM Project
- Guide to KIM IDs
- The OpenKIM Brand Guide
- Licensing
- Team Members
- Governance

KIM API

- Overview
- Download
- OS Setup for the KIM API
- Boot Camp
- Help
- Features

Resources

- Frequently Asked Questions
- Software and Projects using KIM
- KIM Publications
- News Archive
- Reading List
- Requirements Document

OpenKIM Repository

- KIM Items
- Visualizations
- Pipeline
- Pipeline — Query
- kimspec.edn Content Guide
- Properties
- KIM Properties Framework
- EDN Format

Other

- Forums
- Contact Us

The OpenKIM project is supported by NSF funding

The OpenKIM project collaborates with the NIST IPRP

OpenKIM with LAMMPS

OpenKIM Ryan

OS Setup for the KIM API - Overview

Overview | Ubuntu | Macintosh | Windows

Description

These pages provide simple steps to setup the Ubuntu, Mac, and Window Operating Systems (OS's) for use with the KIM API. In addition, instructions for installing these OS's as Virtual Machines (VMs) within VirtualBox (VB) are provided.

If you simply want to make your OS ready for use with the KIM API, click on the appropriate OS tab above and follow the instructions.

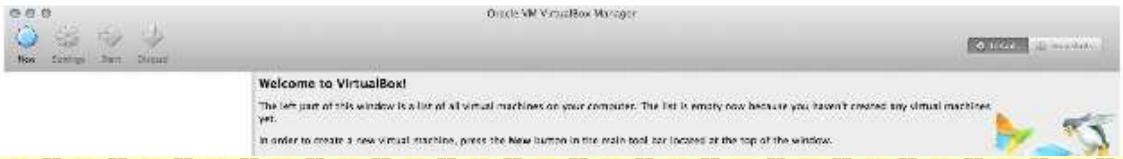
If you want to create a VM with your favorite OS, you will first need to install VirtualBox. See below.

VirtualBox Setup

VirtualBox (VB) is a great tool that allows virtualization. That is, running another OS on the host machine in a window on the host desktop. To download and to find out more about VirtualBox click here.

Starting up VB

Once you have downloaded and installed VB, start it up. It will open up screen that looks like this:



OpenKIM with LAMMPS

OpenKIM Ryan

OS Setup for the KIM API - Ubuntu

Overview | Ubuntu | Macintosh | Windows

Contents:

Downloading the disk image

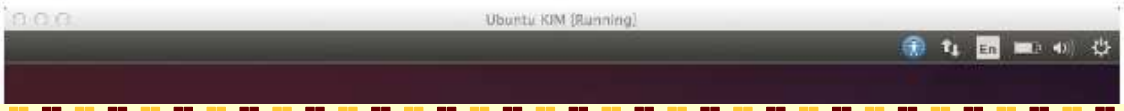
To set up the Ubuntu machine you first need a Ubuntu install disk image. This can be downloaded from here. Choose which version you want (64 bit, 32 bit or 64 bit Mac; 64 bit is recommended for the KIM API). Then click Download.

Creating the Ubuntu Virtual Machine (VM)

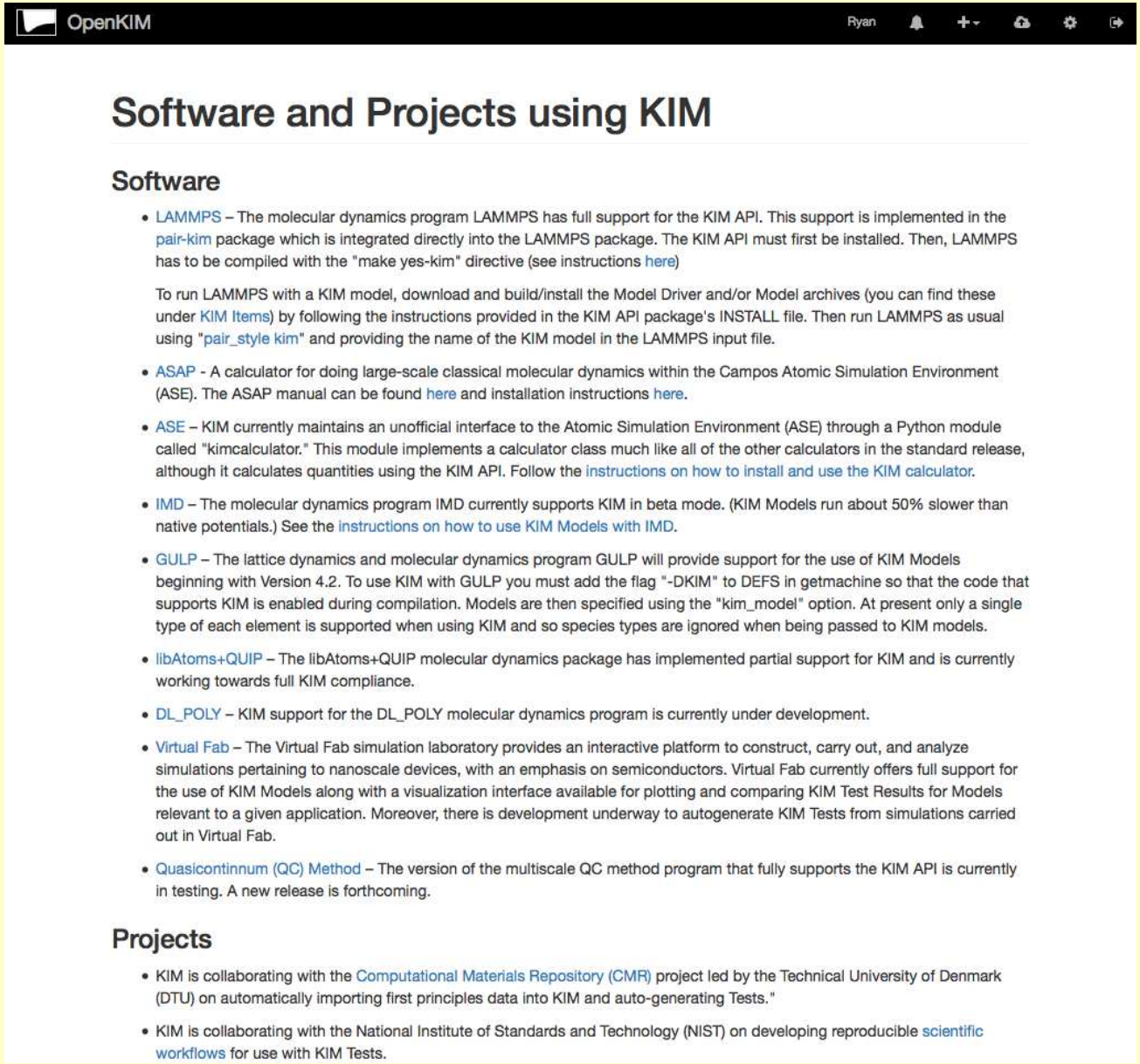
If you want to create a Ubuntu VM follow these instructions.

Installing the Ubuntu OS

Using the disk image, boot your machine. You should see this screen:



OpenKIM with LAMMPS



OpenKIM Ryan

Software and Projects using KIM

Software

- LAMMPS – The molecular dynamics program LAMMPS has full support for the KIM API. This support is implemented in the pair-kim package which is integrated directly into the LAMMPS package. The KIM API must first be installed. Then, LAMMPS has to be compiled with the "make yes-kim" directive (see instructions here)

  To run LAMMPS with a KIM model, download and build/install the Model Driver and/or Model archives (you can find these under KIM Items) by following the instructions provided in the KIM API package's INSTALL file. Then run LAMMPS as usual using "pair_style kim" and providing the name of the KIM model in the LAMMPS input file.

- ASAP - A calculator for doing large-scale classical molecular dynamics within the Campos Atomic Simulation Environment (ASE). The ASAP manual can be found here and installation instructions here.

- ASE – KIM currently maintains an unofficial interface to the Atomic Simulation Environment (ASE) through a Python module called "kimcalculator." This module implements a calculator class much like all of the other calculators in the standard release, although it calculates quantities using the KIM API. Follow the instructions on how to install and use the KIM calculator.

- IMD – The molecular dynamics program IMD currently supports KIM in beta mode. (KIM Models run about 50% slower than native potentials.) See the instructions on how to use KIM Models with IMD.

- GULP – The lattice dynamics and molecular dynamics program GULP will provide support for the use of KIM Models beginning with Version 4.2. To use KIM with GULP you must add the flag "-DKIM" to DEFS in getmachine so that the code that supports KIM is enabled during compilation. Models are then specified using the "kim_model" option. At present only a single type of each element is supported when using KIM and so species types are ignored when being passed to KIM models.

- libAtoms+QUIP – The libAtoms+QUIP molecular dynamics package has implemented partial support for KIM and is currently working towards full KIM compliance.

- DL_POLY – KIM support for the DL_POLY molecular dynamics program is currently under development.

- Virtual Fab – The Virtual Fab simulation laboratory provides an interactive platform to construct, carry out, and analyze simulations pertaining to nanoscale devices, with an emphasis on semiconductors. Virtual Fab currently offers full support for the use of KIM Models along with a visualization interface available for plotting and comparing KIM Test Results for Models relevant to a given application. Moreover, there is development underway to autogenerate KIM Tests from simulations carried out in Virtual Fab.

- Quasicontinnum (QC) Method – The version of the multiscale QC method program that fully supports the KIM API is currently in testing. A new release is forthcoming.

Projects

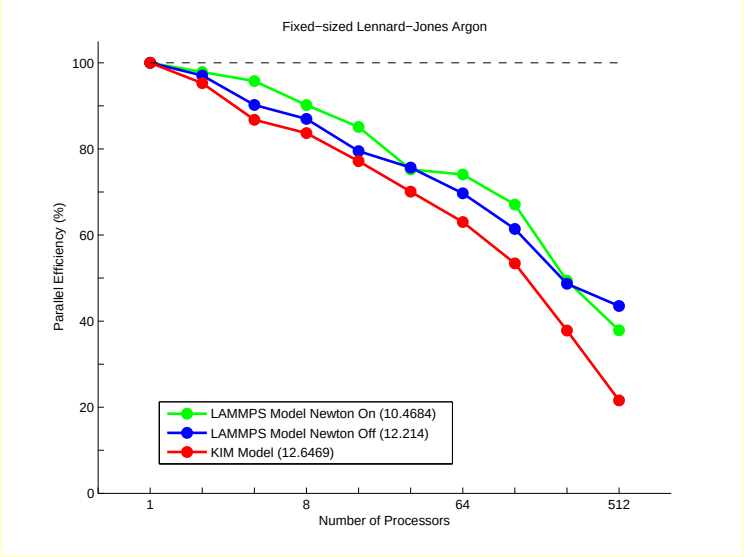
- KIM is collaborating with the Computational Materials Repository (CMR) project led by the Technical University of Denmark (DTU) on automatically importing first principles data into KIM and auto-generating Tests."

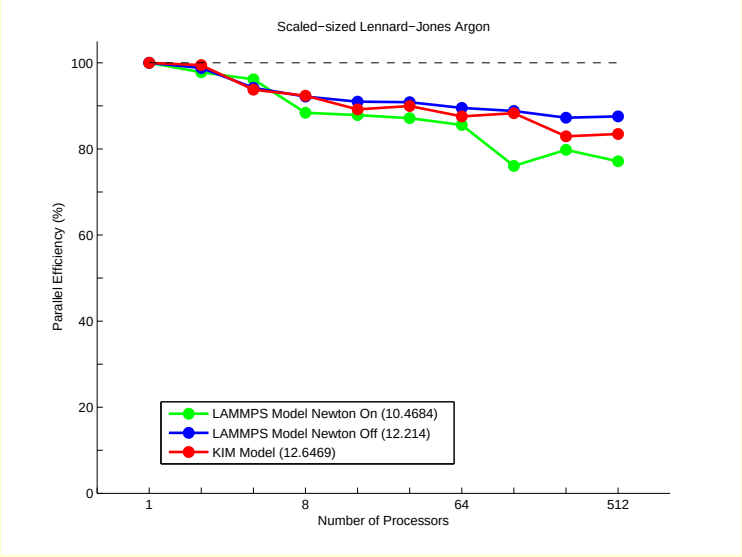
- KIM is collaborating with the National Institute of Standards and Technology (NIST) on developing reproducible scientific workflows for use with KIM Tests.

OpenKIM with LAMMPS

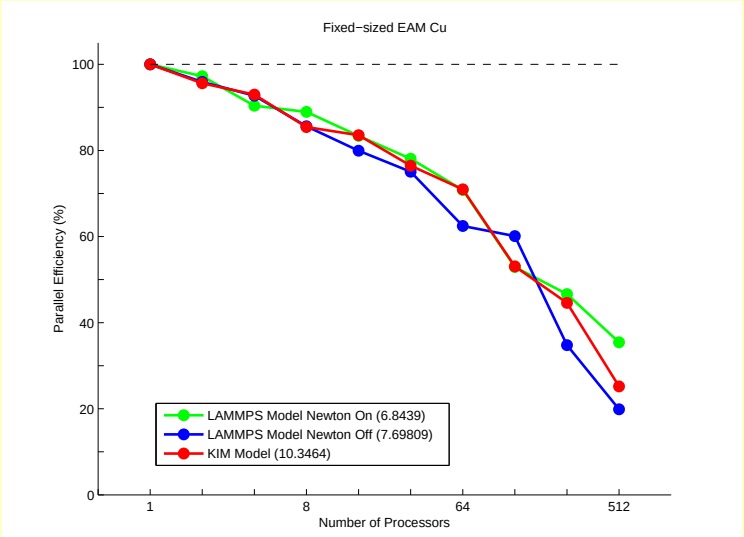
Engineering AEM Aerospace & Mechanics
UNIVERSITY OF MINNESOTA

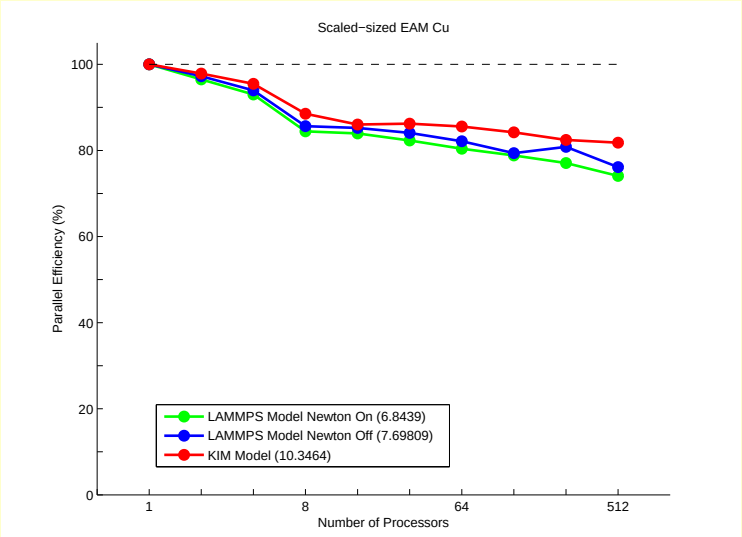
LennardJones612_Universal__MO_826355984548_000





EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000

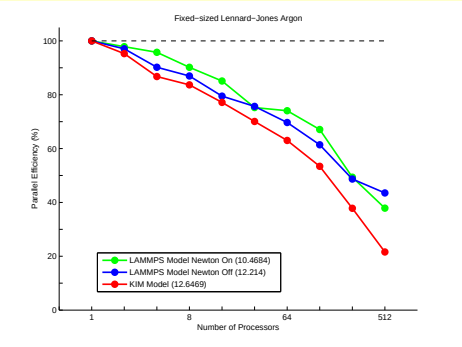


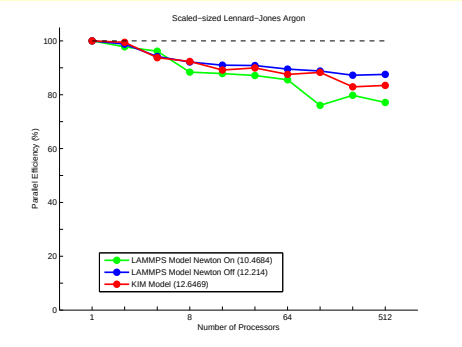


OpenKIM with LAMMPS

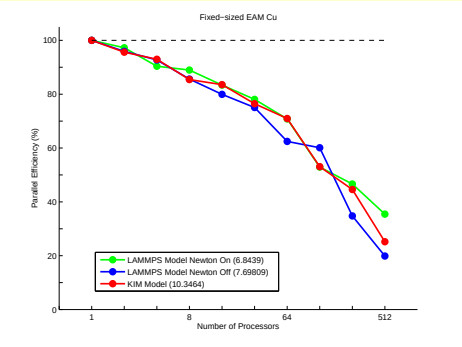
Engineering Aerospace A E & M Mechanics
University of Minnesota

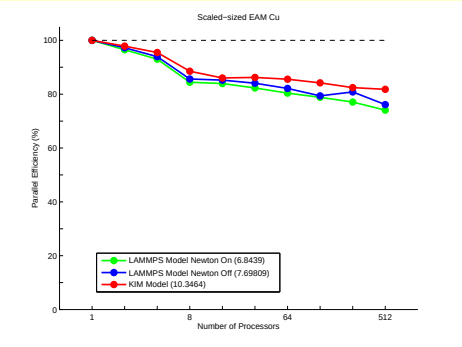
LennardJones612_Universal__MO_826355984548_000





EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000

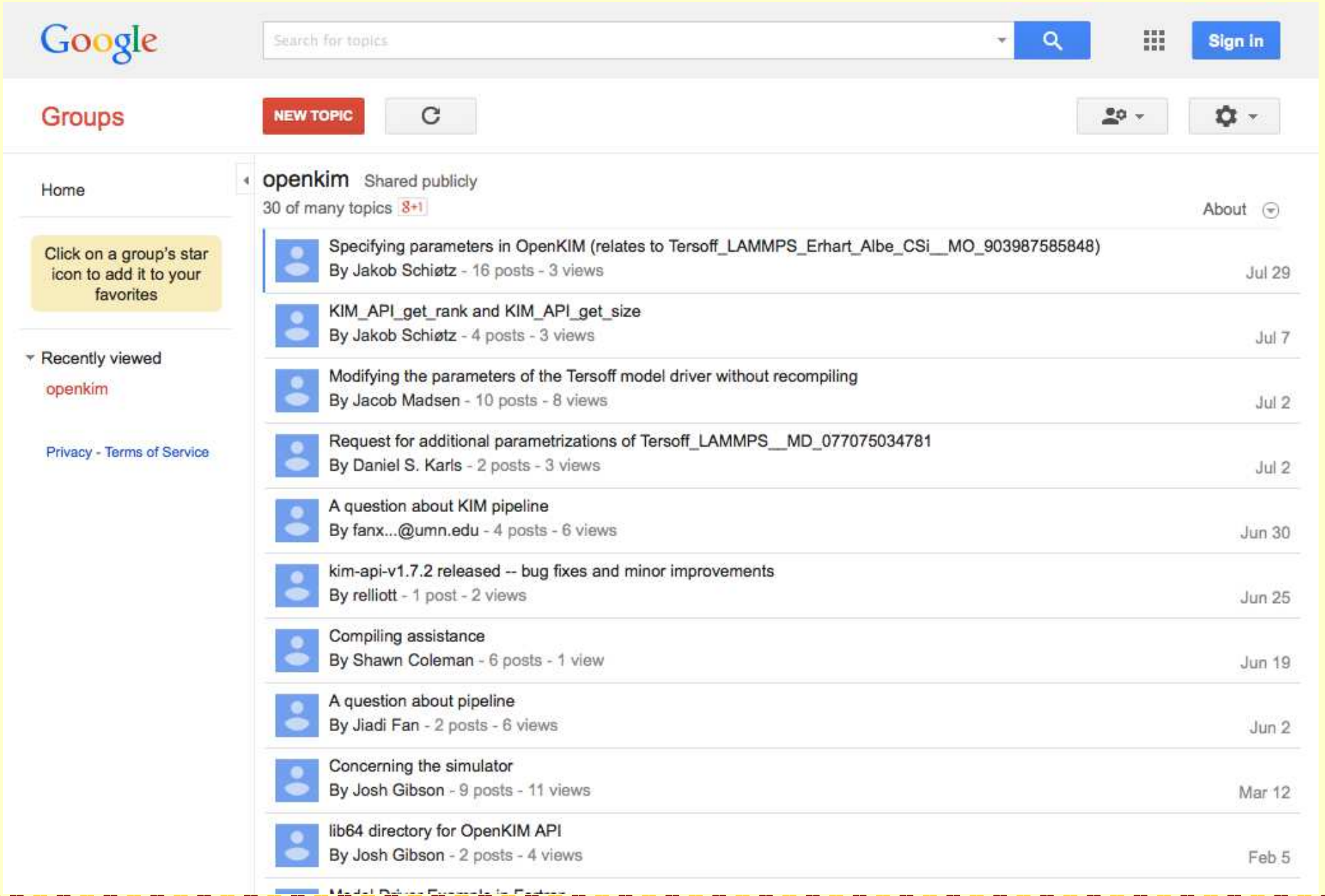




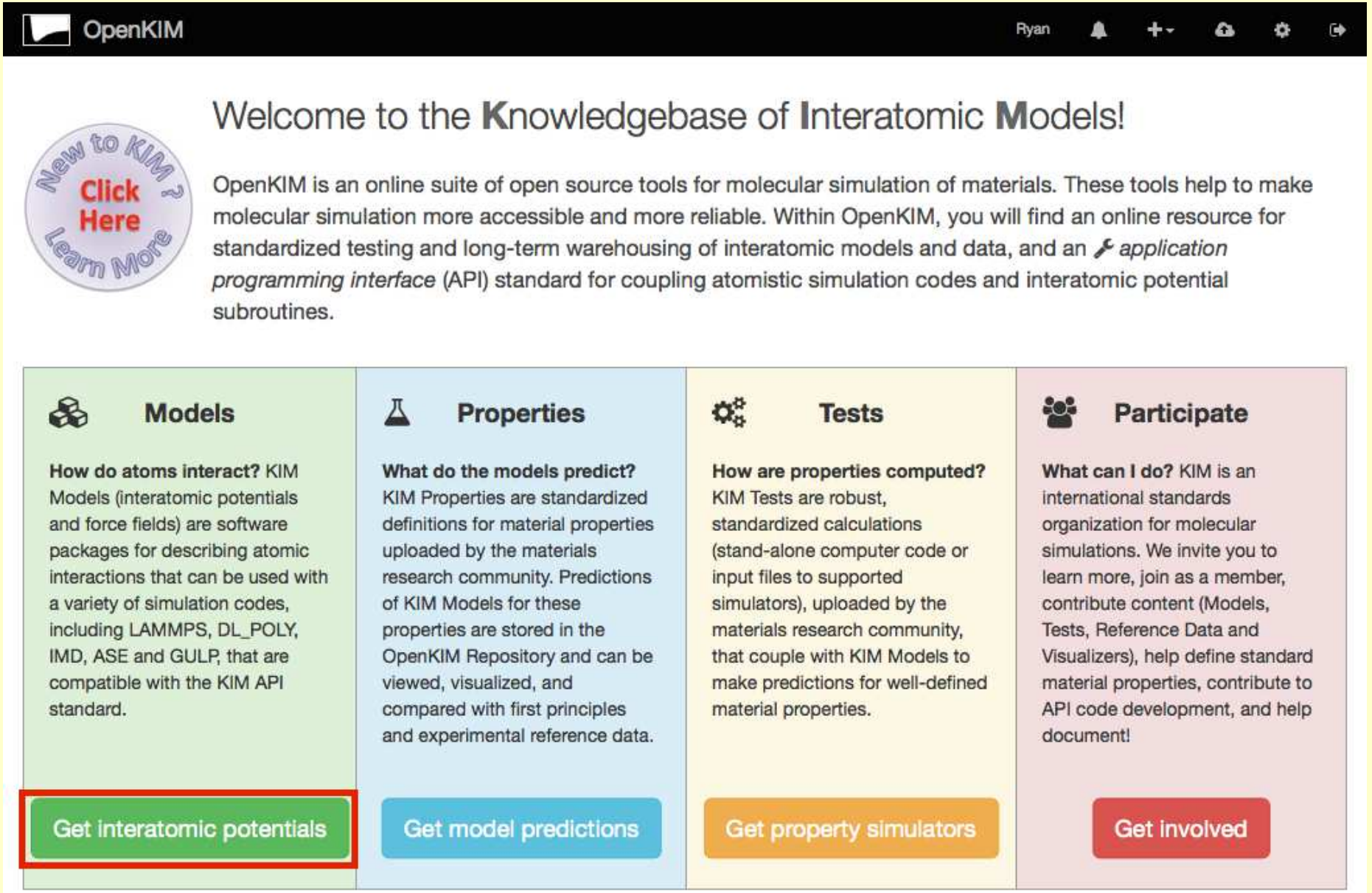
Compiled using the Intel compiler suite (13.1.3) and -O3 optimization and executed on the Minnesota Supercomputing Institute's Itasca machine:

Itasca is an HP Linux cluster with 1,091 HP ProLiant BL280c G6 blade servers, each with two-socket, quad-core 2.8 GHz Intel Xeon X5560 "Nehalem EP" processors sharing 24 GB of system memory, with a 40-gigabit QDR InfiniBand (IB) interconnect. In total, Itasca consists of 8,728 compute cores and 24 TB of main memory.

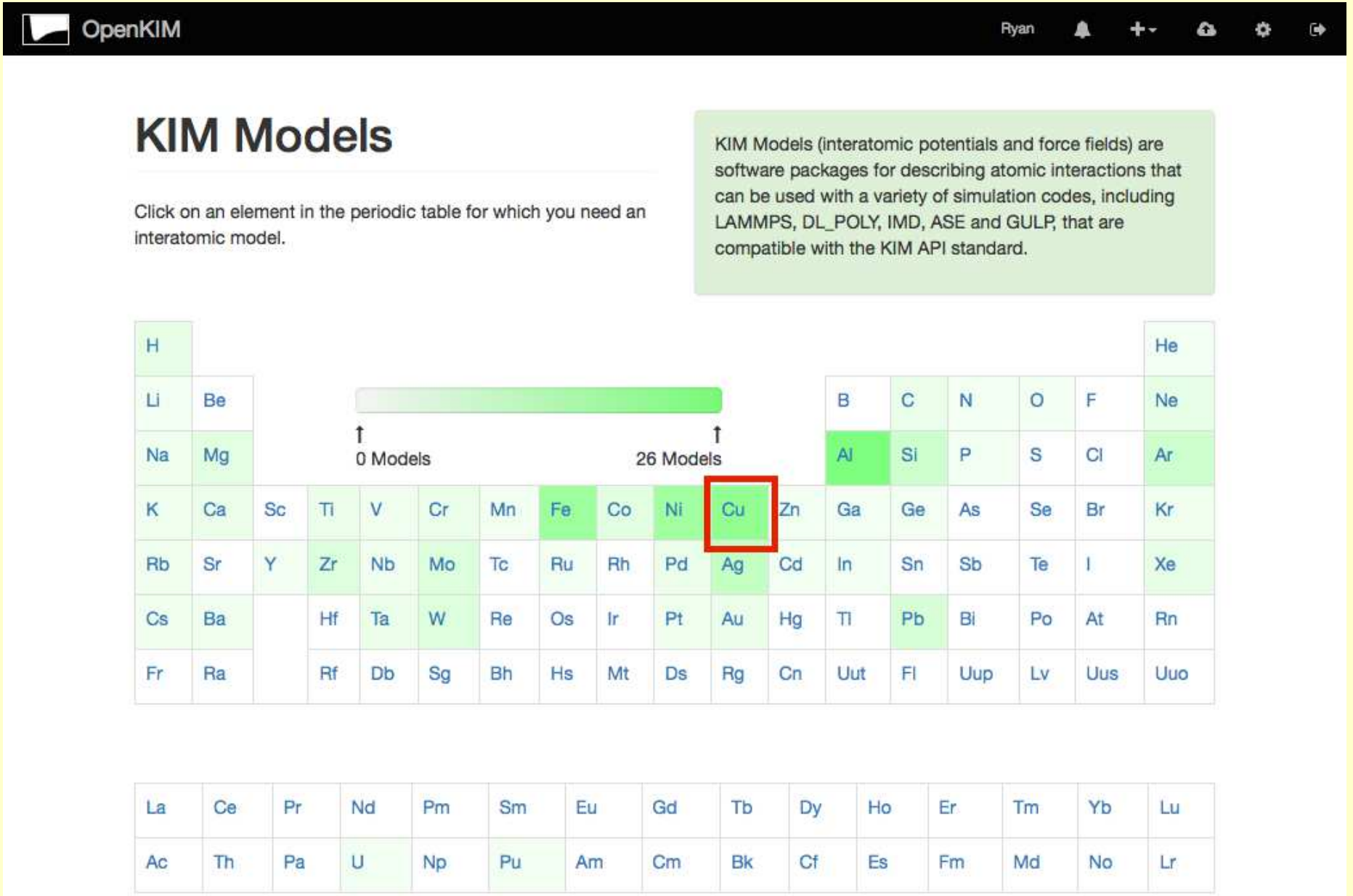
OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS

Cu

Extended KIM ID	Title
EAM_Dynamo_Ackland_Tichy_Cu__MO_179025990738_001	Finnis Sinclair potential for Cu
EAM_Dynamo_Bonny_Pasianot_FeCuNi__MO_469343973171_001	FeCuNi potential to model reactor pressure vessel steels
EAM_Dynamo_Cai_Ye_AlCu__MO_942551040047_001	EAM potential for Al-Cu binary system
EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000	Third universal Cu potential of Foiles, Baskes, and Daw; obtained from LAMMPS
EAM_Dynamo_Hoyt_Garvin_PbCu__MO_119135752160_001	Embedded Atom Method parametrization of the Pb-Cu system
EAM_Dynamo_Mendelev_King_Cu__MO_748636486270_001	FS potential for Cu
EAM_Dynamo_Mendelev_Kramer_Cu__MO_945691923444_001	FS/EAM potential for Cu
EAM_Dynamo_Mendelev_Kramer_CuZr__MO_600021860456_001	FS potential for Cu-Zr
EAM_Dynamo_Mendelev_Sordelet_CuZr__MO_120596890176_001	FS potential for Cu-Zr
EAM_Dynamo_Mishin_Mehl_Cu__MO_346334655118_001	EAM Cu Potential
EAM_Dynamo_Onat_Durukanoglu_CuNi__MO_592013496703_001	An optimized EAM potential for Cu–Ni alloys
EAM_Dynamo_Williams_Mishin_CuAg__MO_128703483589_001	EAM alloy potential for the Cu-Ag system.
EAM_Dynamo_Wu_Trinkle_CuAg__MO_270337113239_001	EAM potential for Cu/Ag(111) Surface Diffusion.
EAM_Dynamo_Zhou_Johnson_Cu__MO_127245782811_001	EAM alloy potential set table, compatible with LAMMPS
EAM_Johnson_NearestNeighbor_Cu__MO_887933271505_001	This is an analytical NN EAM model for Cu by Johnson.
EMT_Asap_MetalGlass_CuMgZr__MO_655725647552_002	Effective Medium Theory potential for CuMg and CuZr alloys, in particular metallic glasses.
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.
MEAM_2NN_Fe_to_Ga__MO_145522277939_001	Model parameterization of 2NN MEAM model
Pair_Morse_Modified_MacDonaldMacDonald_Cu__MO_034823476734_000	Modified Morse pair potential for copper due to MacDonald and MacDonald
Pair_Morse_Shifted_GirifalcoWeizer_HighCutoff_Cu__MO_151002396060_001	This is a Cu Morse Model Parameterization by Girifalco and Weizer using a high accuracy cutoff distance

OpenKIM with LAMMPS

OpenKIM Ryan

Jump to: Tests | Visualizers | Files

Contributor: gbonny (username default)
Maintainer: gbonny (username default)

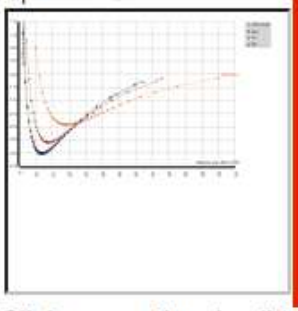
EAM_Dynamo_Bonny_Pasianot_FeCuNi__MO_469343973171_001

Title	FeCuNi potential to model reactor pressure vessel steels
Short KIM ID	MO_469343973171_001
Extended KIM ID	EAM_Dynamo_Bonny_Pasianot_FeCuNi__MO_469343973171_001
KIM Item Type	Parameterized Model using Model Driver EAM_Dynamo__MD_120291908751_001
Species	Cu, Fe, Ni
Description	Ternary FeCuNi EAM-type potential. The fitting was focussed on solute-point defect interaction in the bcc Fe matrix. With respect to thermodynamics the following was accounted for: experimentally observed intermetallic compounds in the FeNi alloys, the Cu solubility in the FeCu binary and the CuNi miscibility gap. The potential is designed to model radiation damage in the FeCuNi model alloy which represents reactor pressure vessel steels. FeNi cross potential is taken from [Bonny et. al., Modelling Simul. Mater. Sci. Eng. 17 (2009) 025010]. FeCu cross potential is taken from [Pasianot and Malerba, J. Nucl. Mater. 360 (2007) 118]. Fe potential is taken from [Mendelev et al., Philos. Mag. 83 (2003) 3977]. Ni potential is taken from [Voter and Chen, Mater. Res. Soc. Symp. Proc. 82 (1987) 175]. Cu potential is taken from [Mishin et al., Phys. Rev. B 63 (2001) 224106].
Disclaimer	The potential is stiffened.
Source Citations	FeCuNi potential: G. Bonny, R.C. Pasianot, N. Castin and L. Malerba, Philos. Mag. 89 (2009) 3531

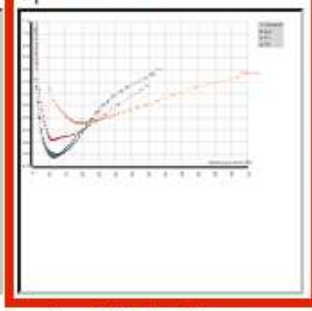
Cohesive Energy Graph

This graph shows the cohesive energy versus volume-per-atom for the current mode 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.) Graphs are generated for each species supported by the model.

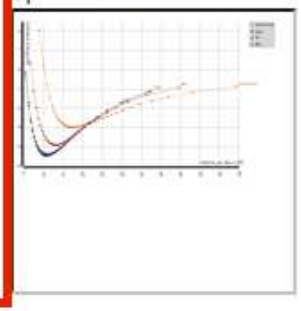
Species: Cu



Species: Fe



Species: Ni

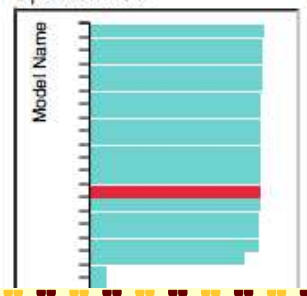


Click on any thumbnail to get a full size image.

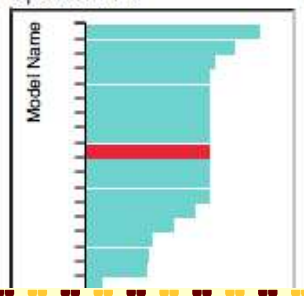
Horizontal Chart

This bar chart plot shows the mono-atomic face-centered cubic (fcc) lattice constant predicted by the current model (shown in red) compared with the predictions for all other models in the OpenKIM Repository that support the species. The vertical bars show the average and standard deviation (one sigma) bounds for all model predictions. Graphs are generated for each species supported by the model.

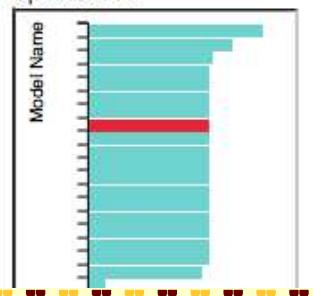
Species: Cu



Species: Fe



Species: Ni

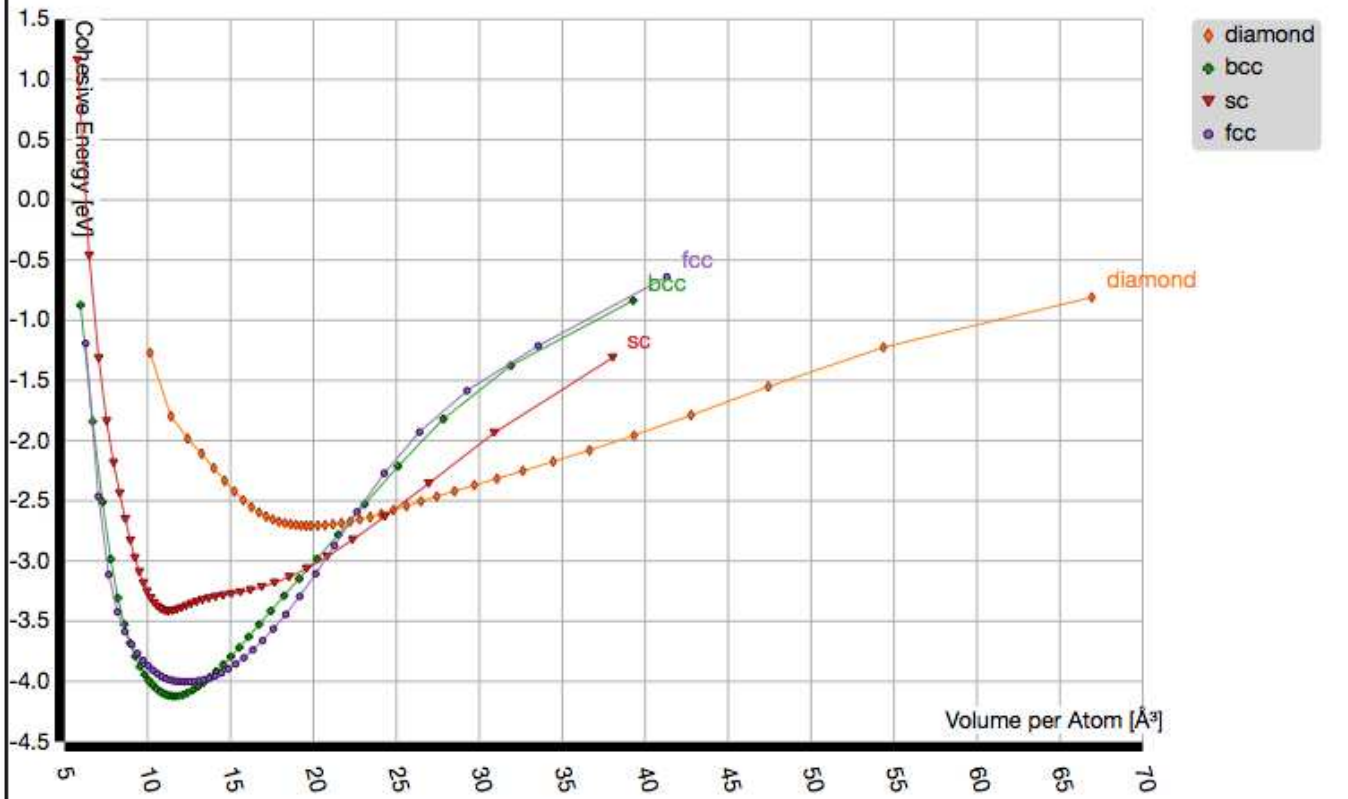


OpenKIM with LAMMPS

OpenKIM Ryan

Model: EAM_Dynamo_Bonny_Pasianot_FeCuNi__MO_469343973171_001

Species: Fe



OpenKIM with LAMMPS

Cu

Extended KIM ID	Title
EAM_Dynamo_Ackland_Tichy_Cu__MO_179025990738_001	Finnis Sinclair potential for Cu
EAM_Dynamo_Bonny_Pasianot_FeCuNi__MO_469343973171_001	FeCuNi potential to model reactor pressure vessel steels
EAM_Dynamo_Cai_Ye_AlCu__MO_942551040047_001	EAM potential for Al-Cu binary system
EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000	Third universal Cu potential of Foiles, Baskes, and Daw; obtained from LAMMPS
EAM_Dynamo_Hoyt_Garvin_PbCu__MO_119135752160_001	Embedded Atom Method parametrization of the Pb-Cu system
EAM_Dynamo_Mendelev_King_Cu__MO_748636486270_001	FS potential for Cu
EAM_Dynamo_Mendelev_Kramer_Cu__MO_945691923444_001	FS/EAM potential for Cu
EAM_Dynamo_Mendelev_Kramer_CuZr__MO_600021860456_001	FS potential for Cu-Zr
EAM_Dynamo_Mendelev_Sordelet_CuZr__MO_120596890176_001	FS potential for Cu-Zr
EAM_Dynamo_Mishin_Mehl_Cu__MO_346334655118_001	EAM Cu Potential
EAM_Dynamo_Onat_Durukanoglu_CuNi__MO_592013496703_001	An optimized EAM potential for Cu–Ni alloys
EAM_Dynamo_Williams_Mishin_CuAg__MO_128703483589_001	EAM alloy potential for the Cu-Ag system.
EAM_Dynamo_Wu_Trinkle_CuAg__MO_270337113239_001	EAM potential for Cu/Ag(111) Surface Diffusion.
EAM_Dynamo_Zhou_Johnson_Cu__MO_127245782811_001	EAM alloy potential set table, compatible with LAMMPS
EAM_Johnson_NearestNeighbor_Cu__MO_887933271505_001	This is an analytical NN EAM model for Cu by Johnson.
EMT_Asap_MetalGlass_CuMgZr__MO_655725647552_002	Effective Medium Theory potential for CuMg and CuZr alloys, in particular metallic glasses.
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.
MEAM_2NN_Fe_to_Ga__MO_145522277939_001	Model parameterization of 2NN MEAM model
Pair_Morse_Modified_MacDonaldMacDonald_Cu__MO_034823476734_000	Modified Morse pair potential for copper due to MacDonald and MacDonald
Pair_Morse_Shifted_GirifalcoWeizer_HighCutoff_Cu__MO_151002396060_001	This is a Cu Morse Model Parameterization by Girifalco and Weizer using a high accuracy cutoff distance

OpenKIM with LAMMPS

OpenKIM Ryan

Jump to: Tests | Visualizers | Files

Contributor: Ryan
Maintainer: Ryan

EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000

Title	Third universal Cu potential of Foiles, Baskes, and Daw; obtained from LAMMPS
Short KIM ID	MO_666348409573_000
Extended KIM ID	EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000
KIM Item Type	Parameterized Model using Model Driver EAM_Dynamo__MD_120291908751_001
Species	Cu
Description	This Cu EAM potential parameter file was obtained from the LAMMPS distribution and is dated 2007-06-11. It is the "universal 3" potential from the paper by Foiles, Baskes, and Daw.
Disclaimer	
Source Citations	"Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys," by S. M. Foiles, M. I. Baskes, and M. S. Daw, Phys. Rev. B 33, 7983 (1986); Erratum Phys. Rev. B 37, 10378 (1988) http://dx.doi.org/10.1103/PhysRevB.33.7983
Programming Language(s)	N/A

° Visualizers (in-page)

OpenKIM with LAMMPS

Files

- Cu_u3.eam
- LICENSE
- Makefile
- kimspec.edn

Download

EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000.txz	Tar+XZ	Linux and OS X archive (modern compression)
EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000.tgz	Tar+Gzip	Linux and OS X archive (legacy compression)
EAM_Dynamo_Foiles_Baskes_Universal3_Cu__MO_666348409573_000.zip	Zip	Windows archive

⚠ Download Dependency

This Model requires a Model Driver. Archives for the Model Driver EAM_Dynamo__MD_120291908751_001 appear below.

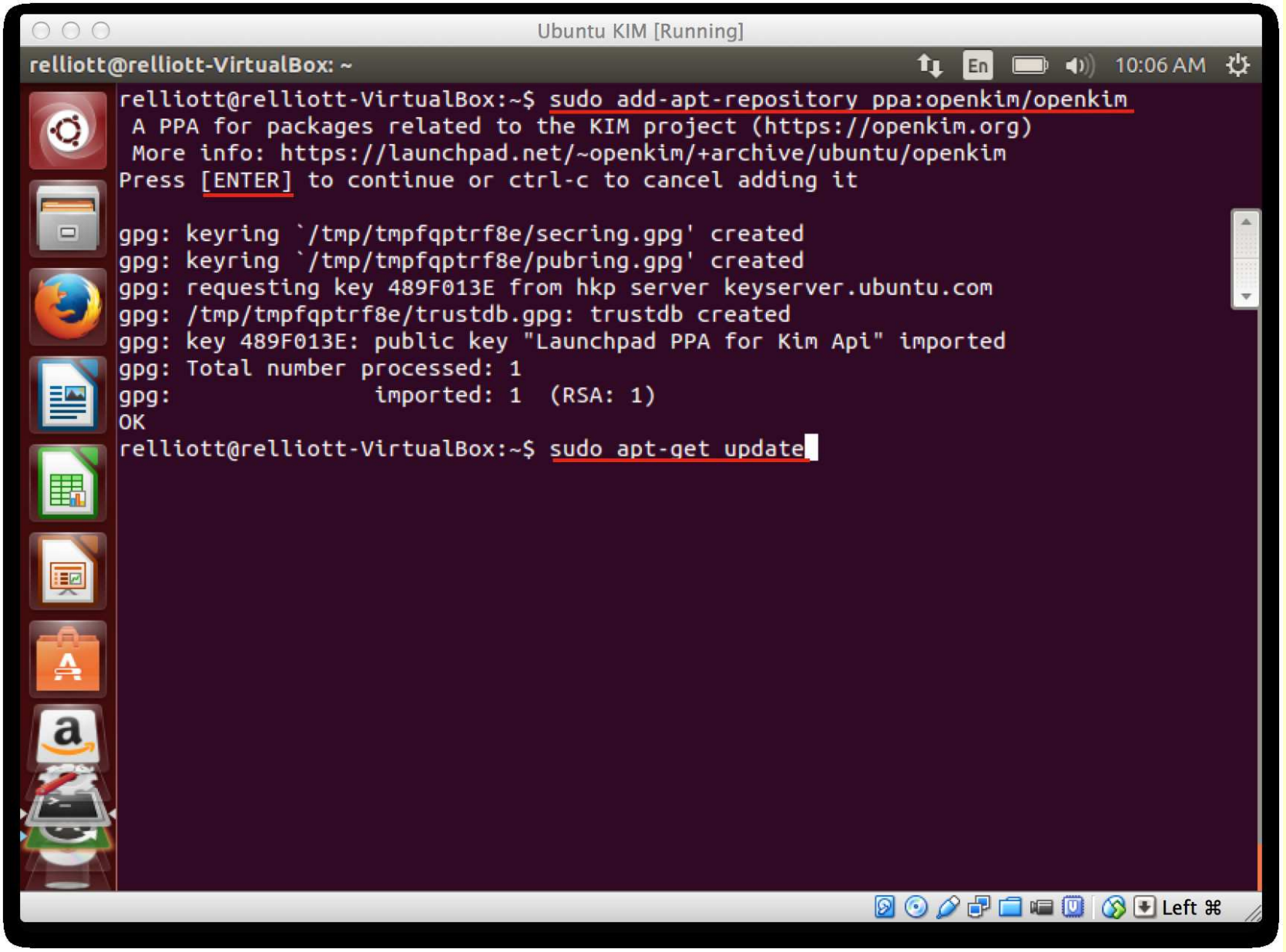
EAM_Dynamo__MD_120291908751_001.txz	Tar+XZ	Linux and OS X archive (modern compression)
EAM_Dynamo__MD_120291908751_001.tgz	Tar+Gzip	Linux and OS X archive (legacy compression)
EAM_Dynamo__MD_120291908751_001.zip	Zip	Windows archive

OUTLINE

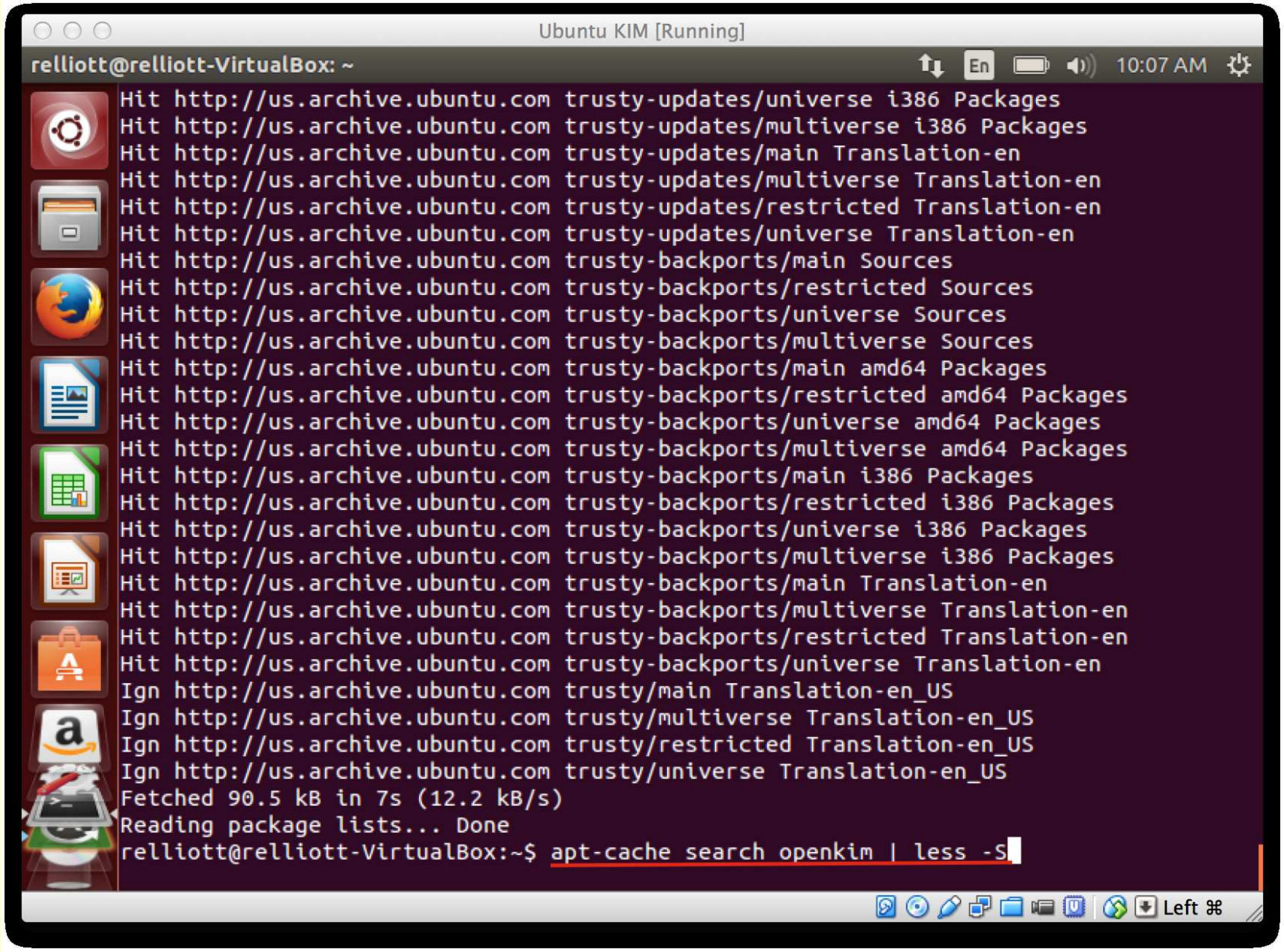
1. The Knowledgebase of Interatomic Models
   - User Signup
   - KIM API
   - "KIM Compliant" Simulators
   - Setup VirtualBox Ubuntu VM
   - KIM Model performance within LAMMPS
   - KIM Model pages
2. Installing the KIM API on Ubuntu (ppa:openkim/openkim)
3. Using LAMMPS with the KIM API

Appendix: Installing the KIM API by hand

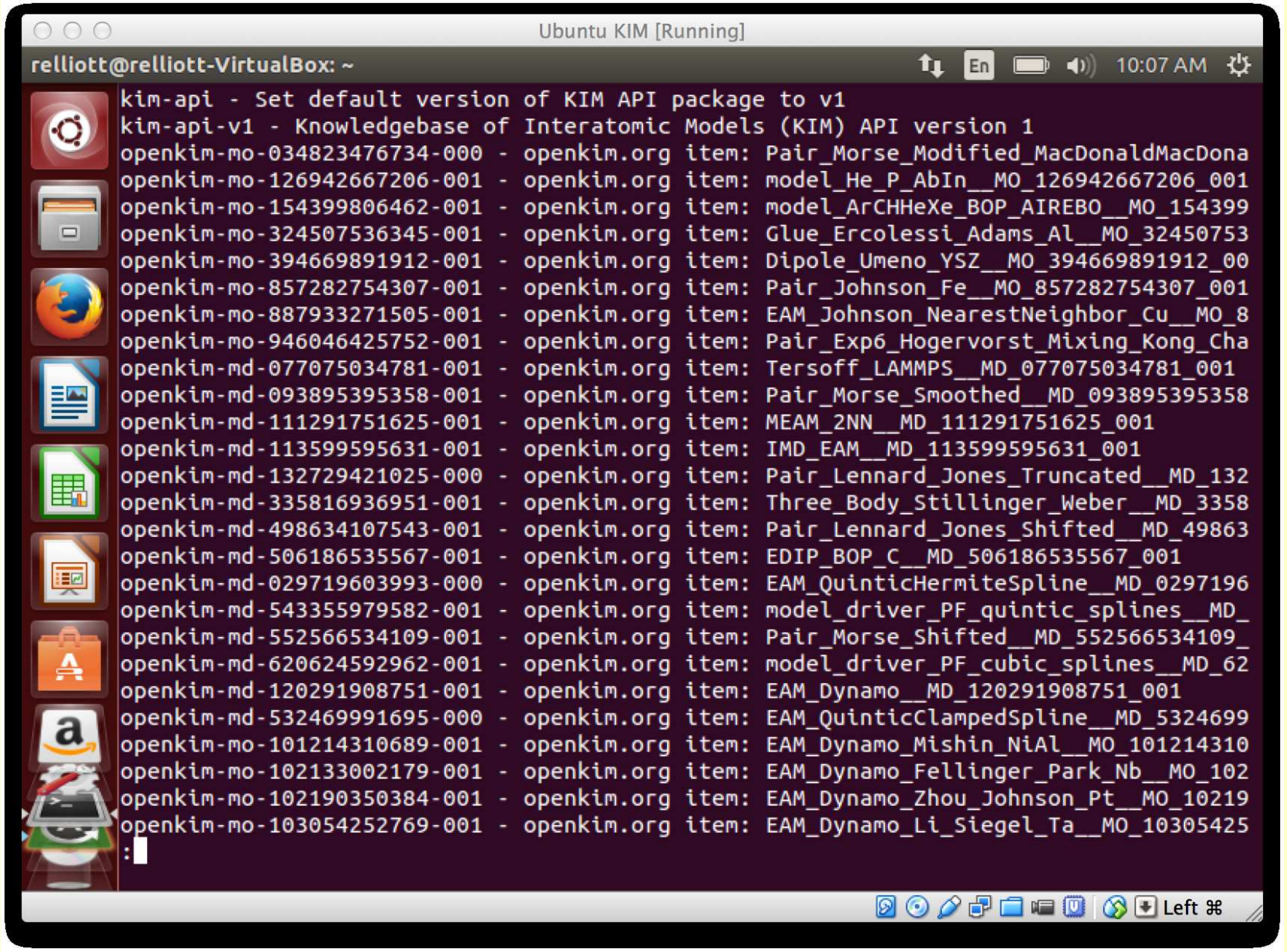
OpenKIM with LAMMPS



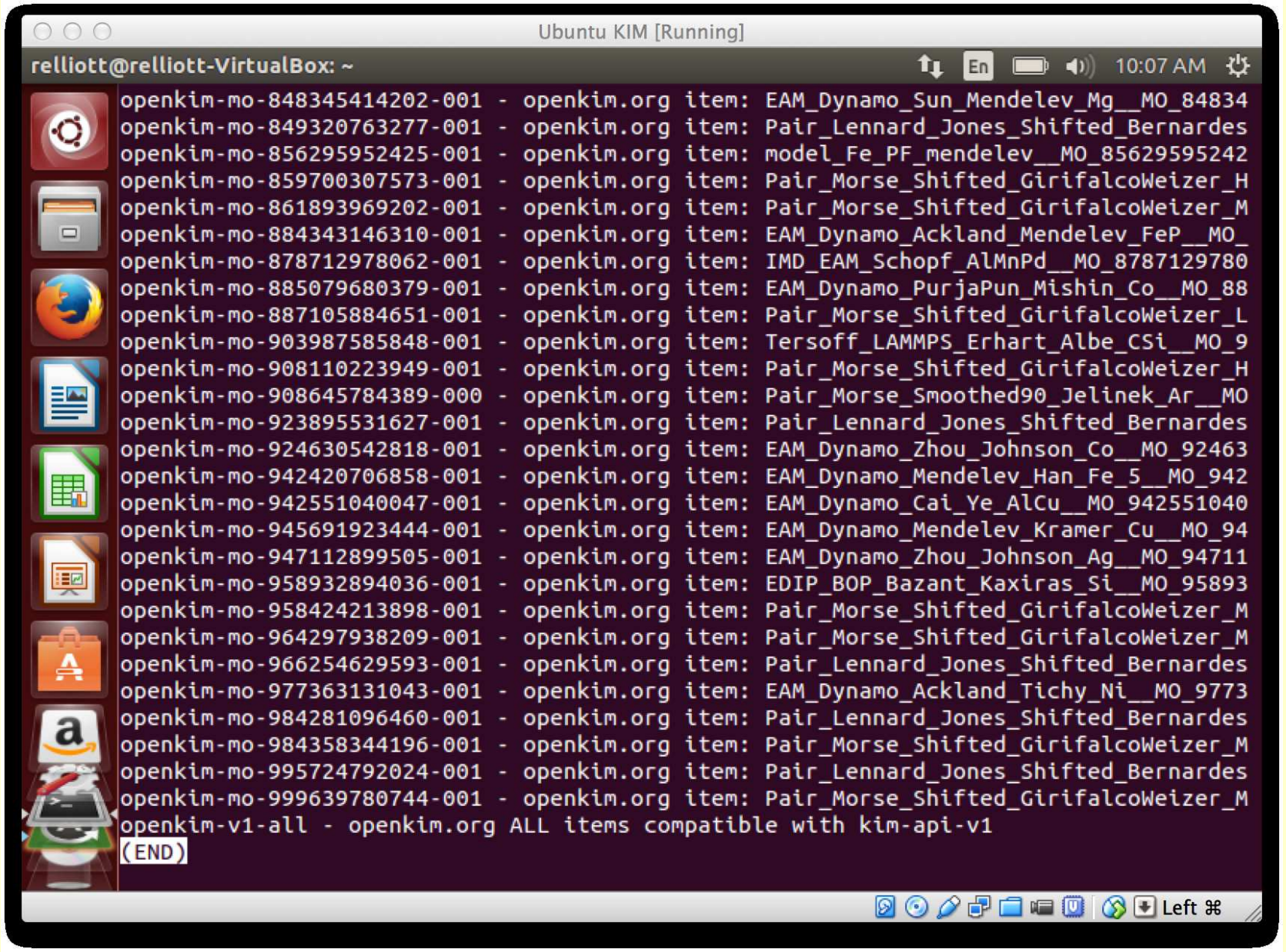
OpenKIM with LAMMPS



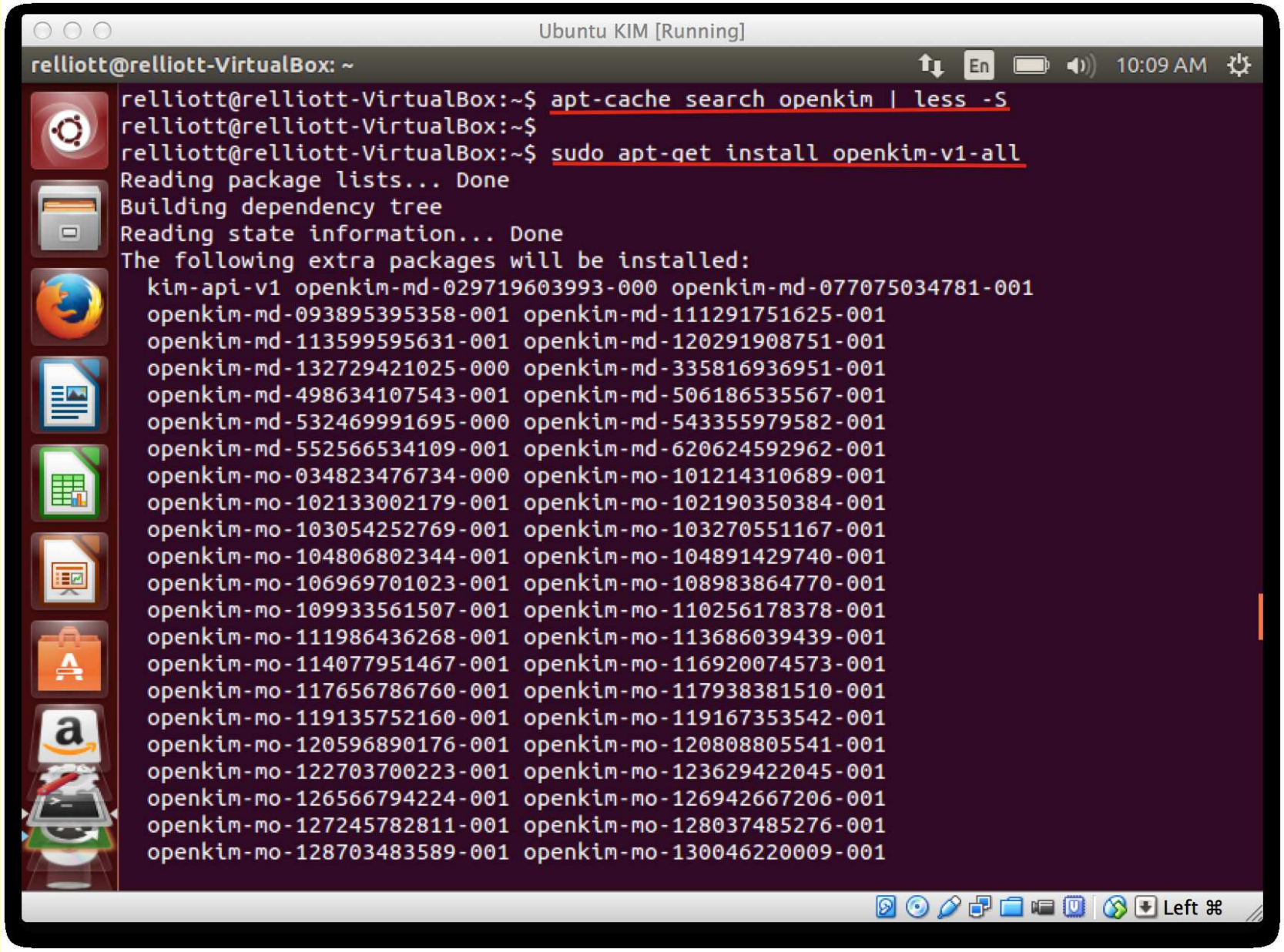
OpenKIM with LAMMPS



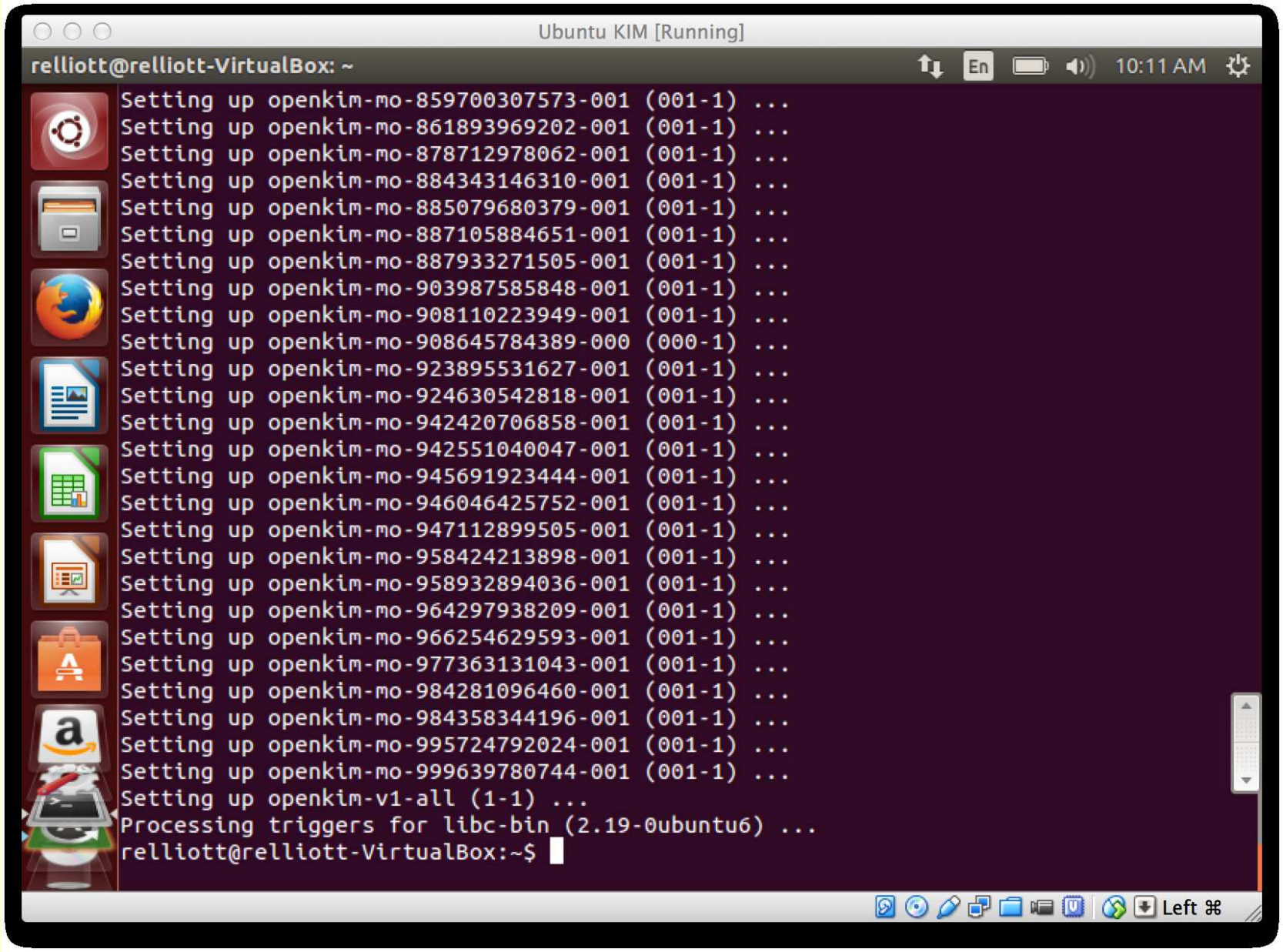
OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS

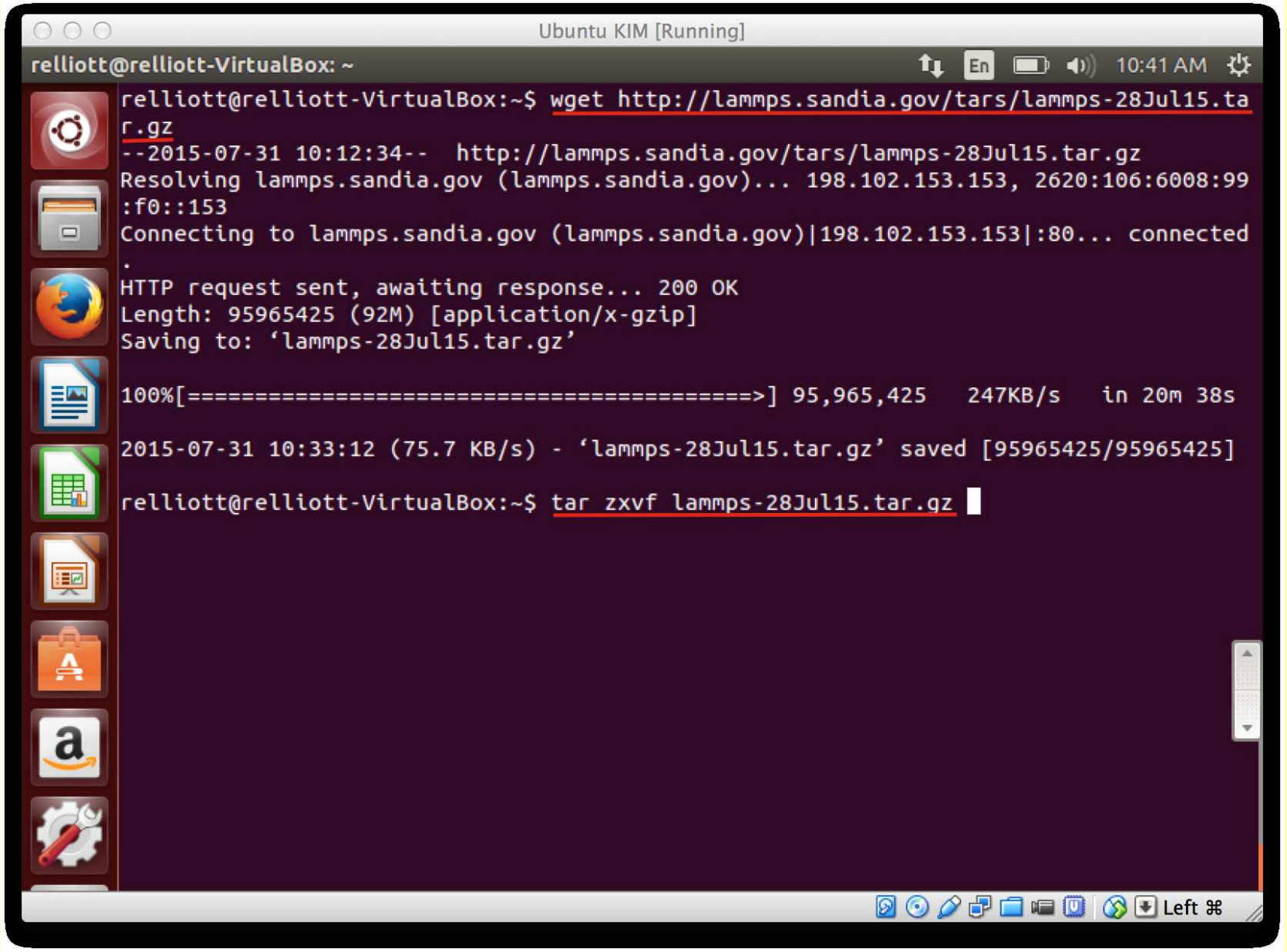


OUTLINE

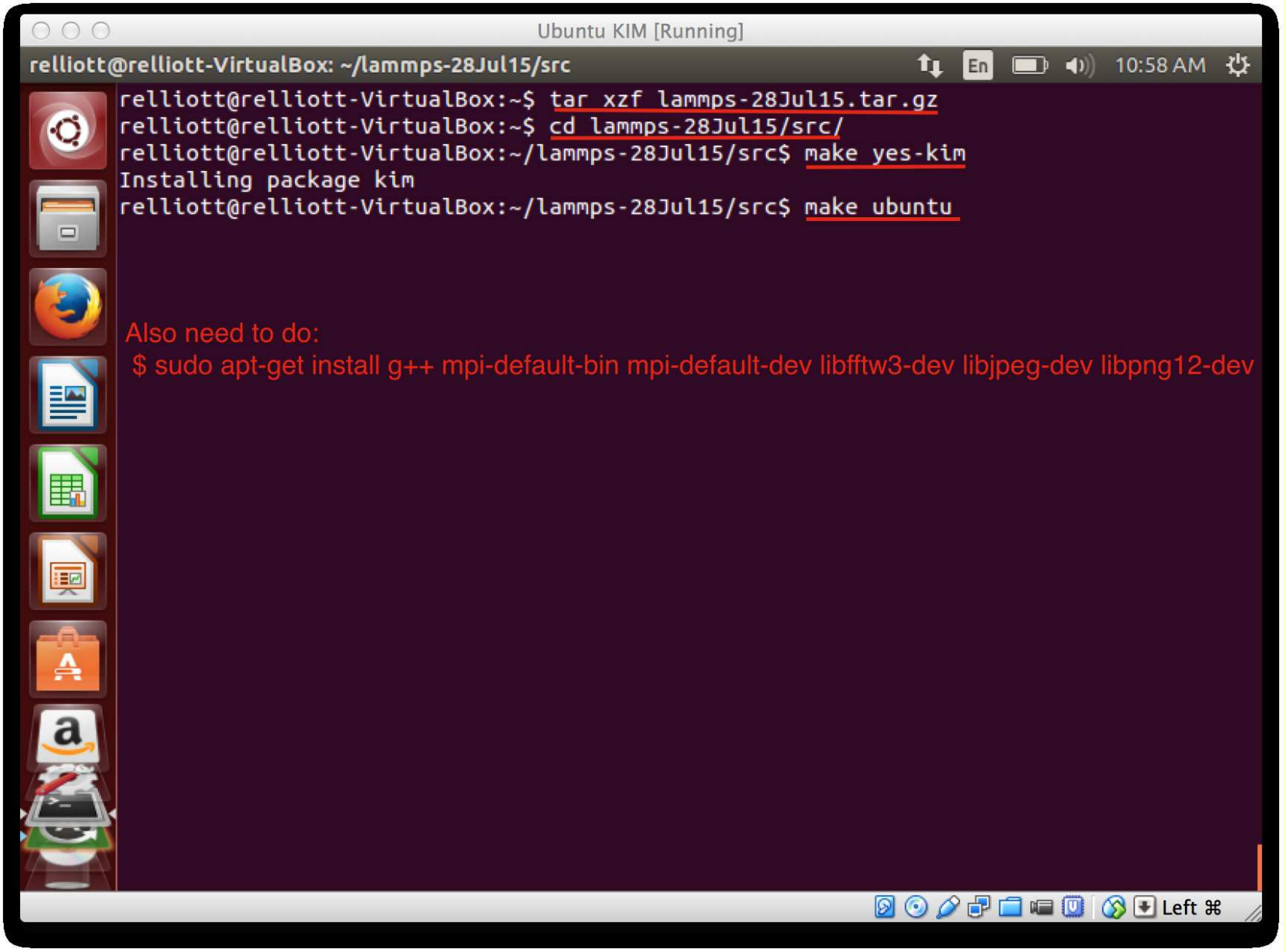
1. The Knowledgebase of Interatomic Models
   - User Signup
   - KIM API
   - "KIM Compliant" Simulators
   - Setup VirtualBox Ubuntu VM
   - KIM Model performance within LAMMPS
   - KIM Model pages
2. Installing the KIM API on Ubuntu (ppa:openkim/openkim)
3. Using LAMMPS with the KIM API

Appendix: Installing the KIM API by hand

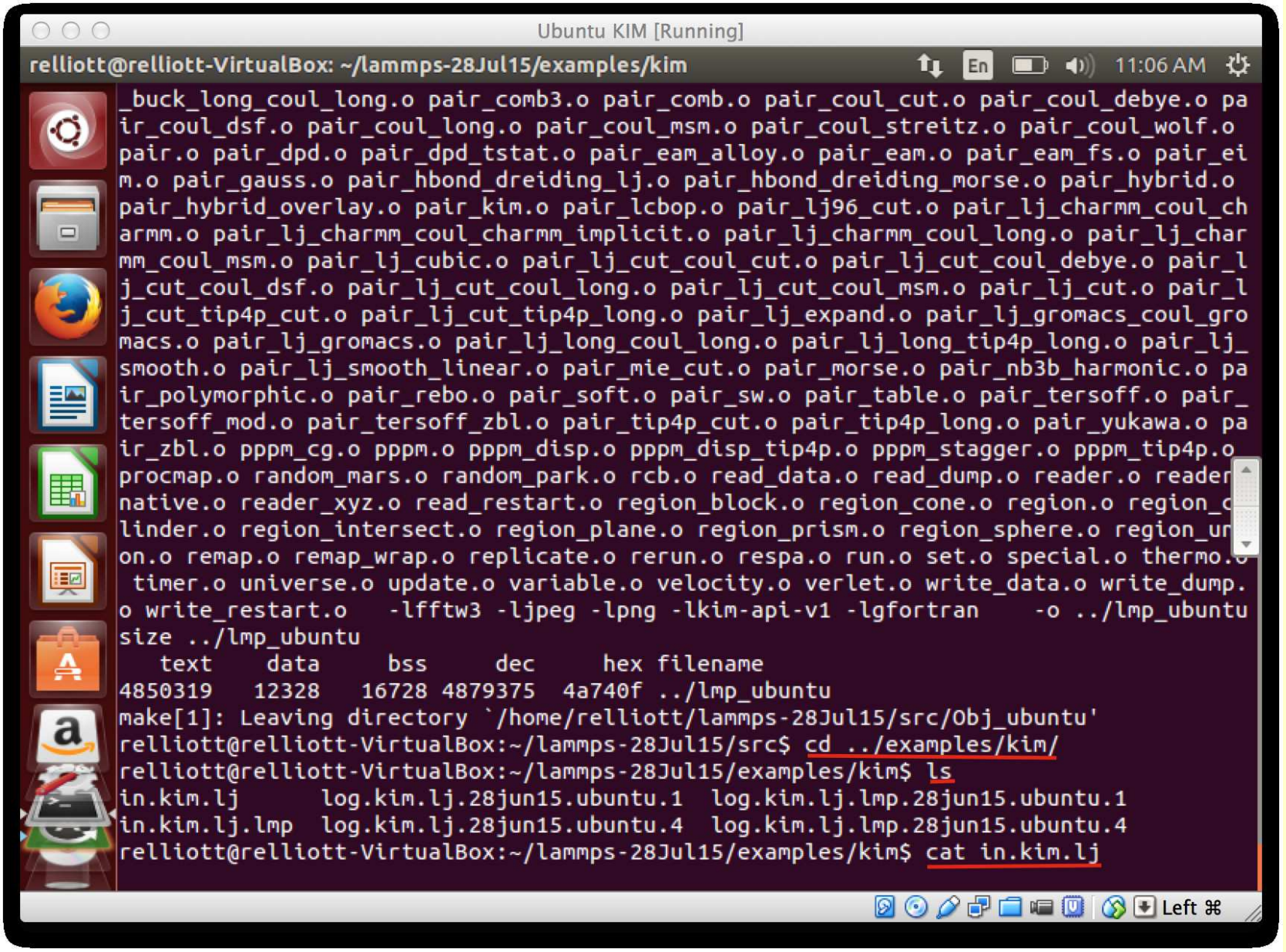
OpenKIM with LAMMPS



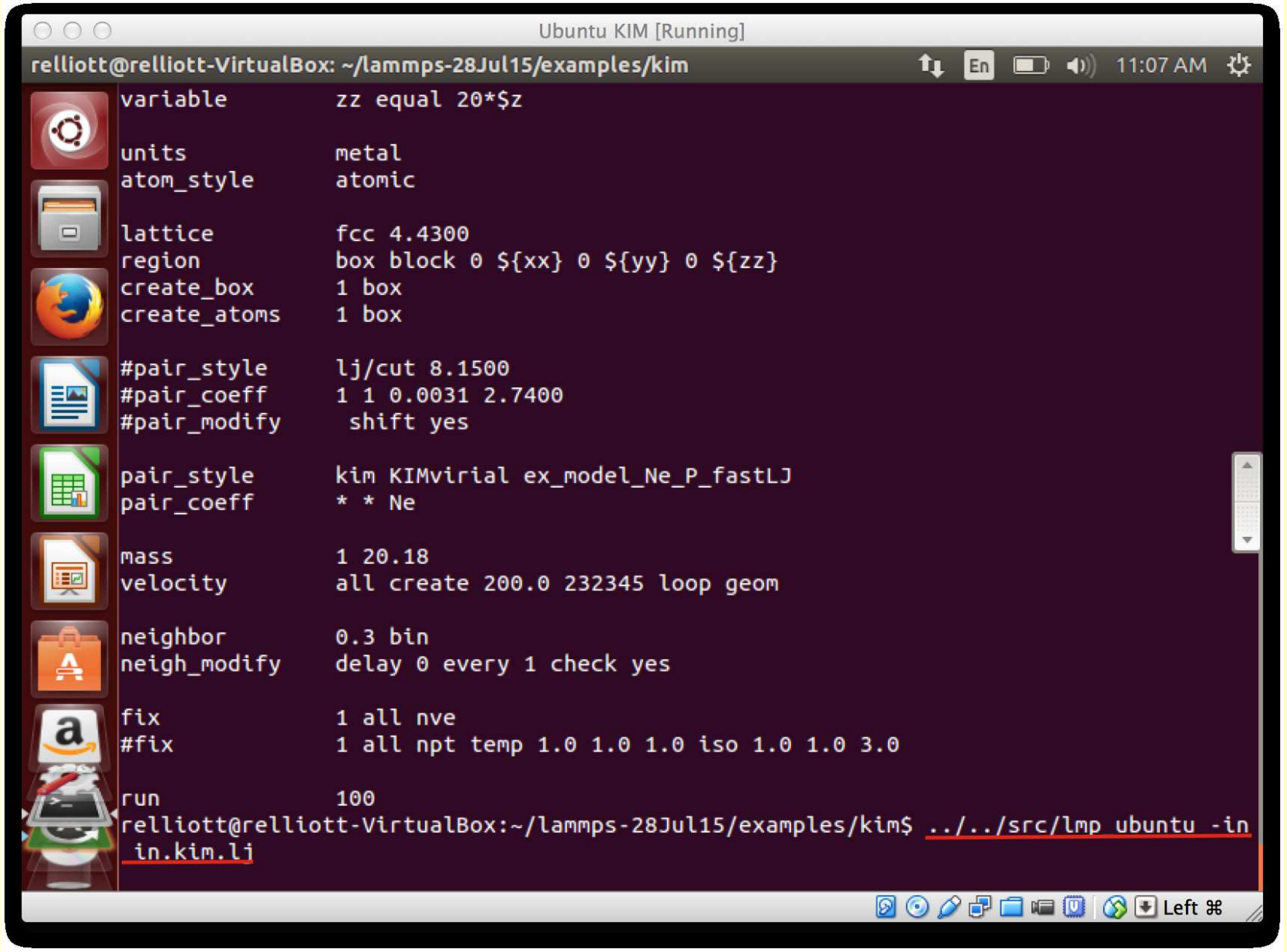
OpenKIM with LAMMPS



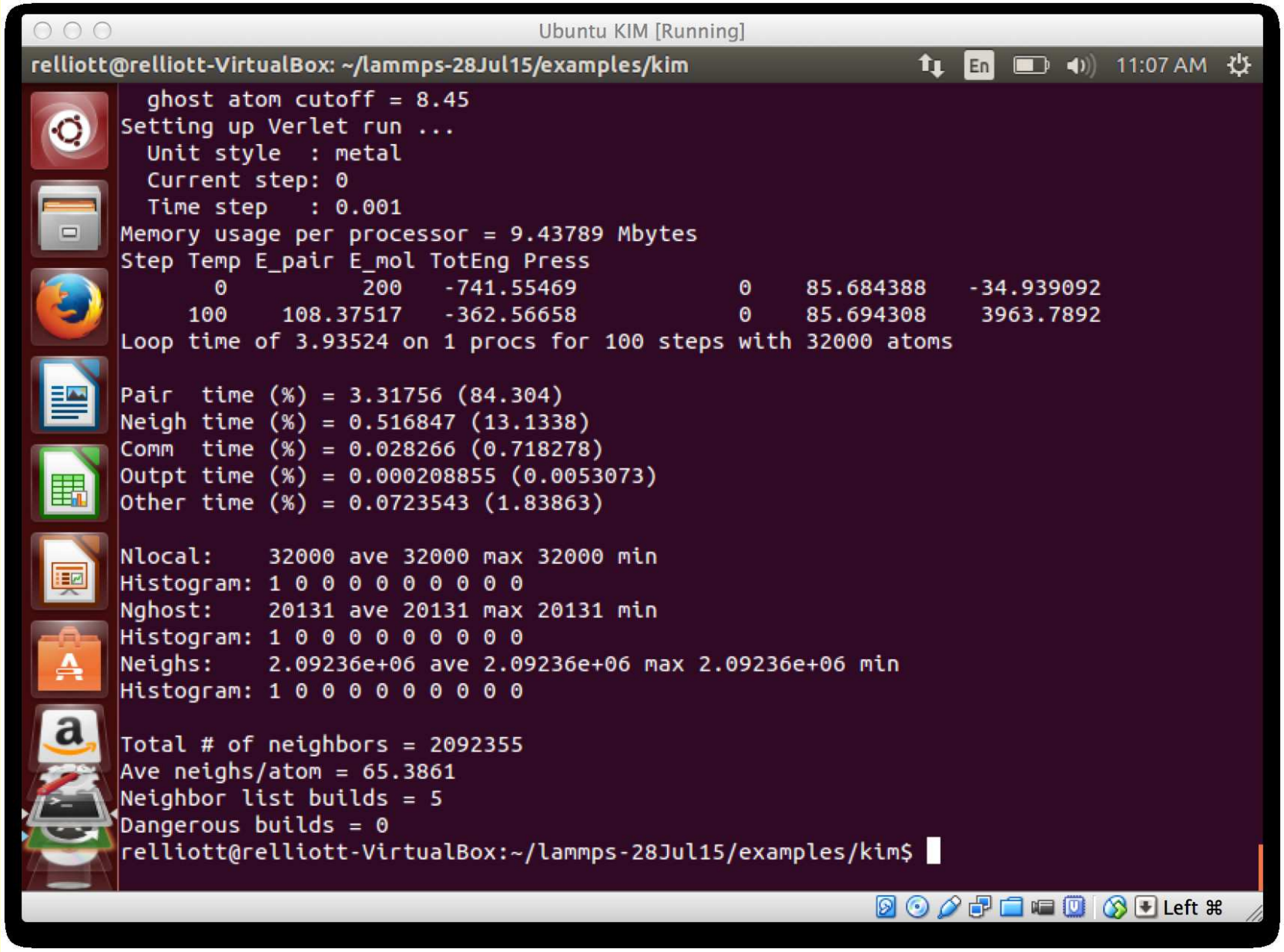
OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS

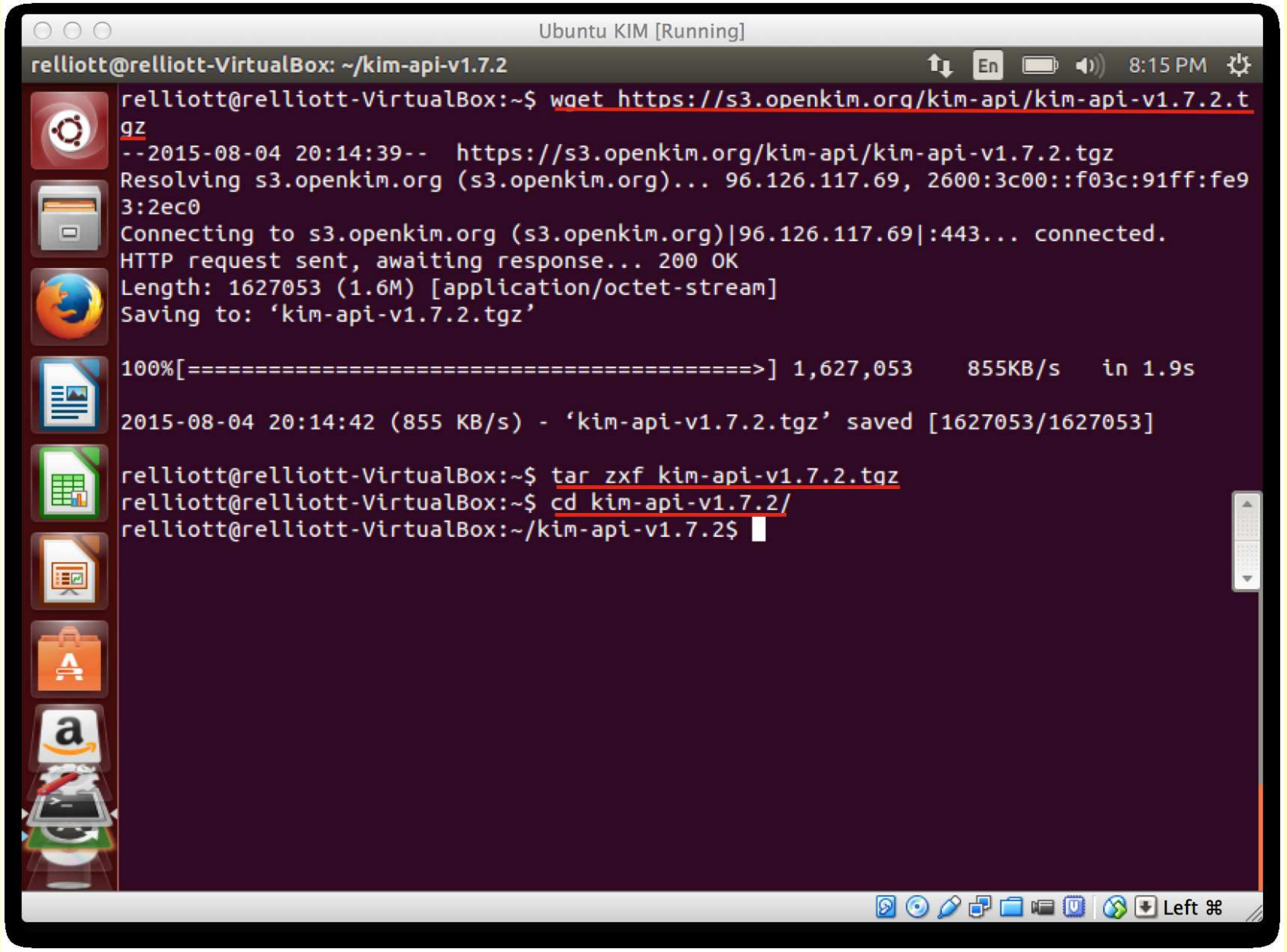


OUTLINE

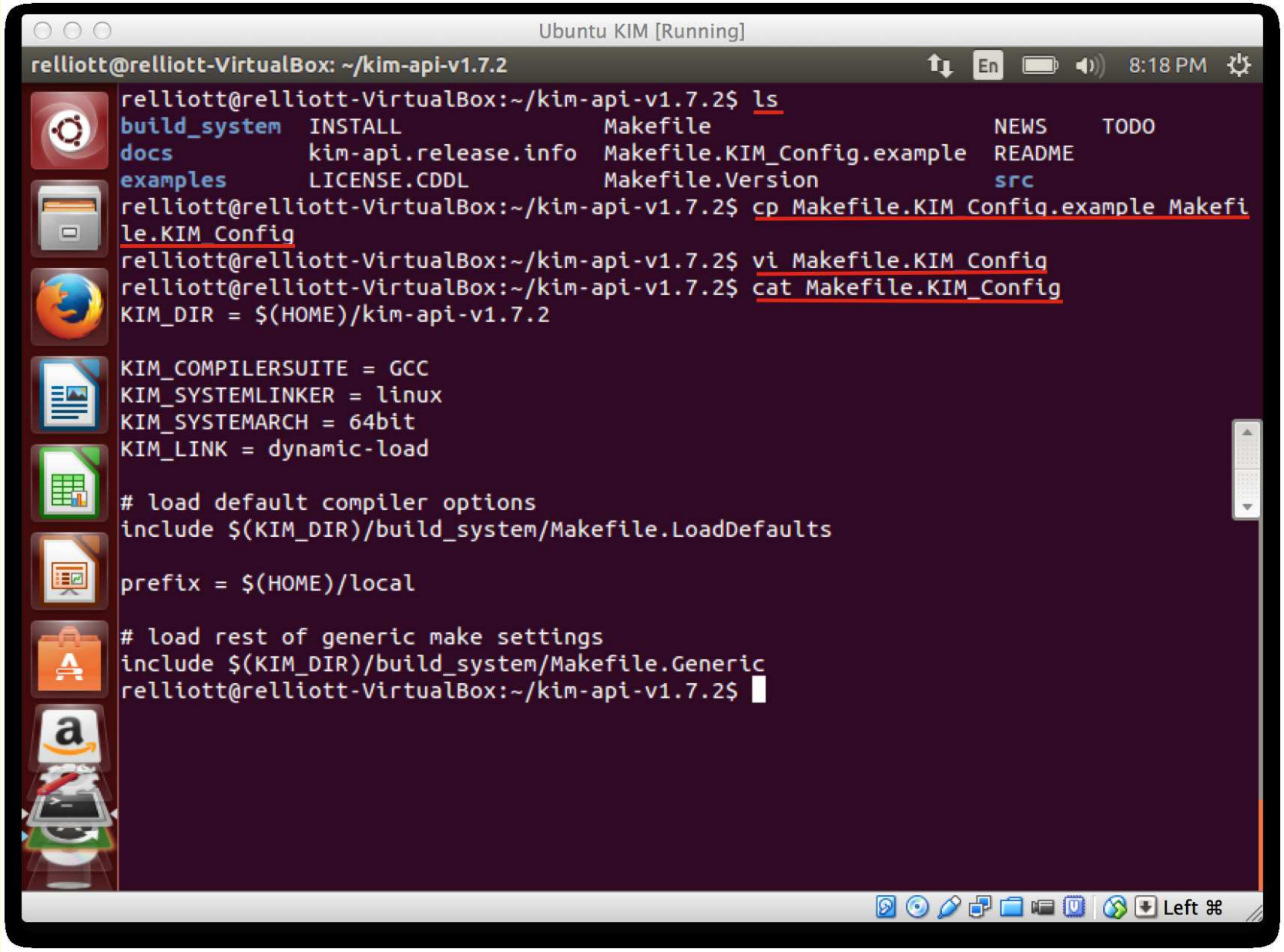
1. The Knowledgebase of Interatomic Models
   - User Signup
   - KIM API
   - "KIM Compliant" Simulators
   - Setup VirtualBox Ubuntu VM
   - KIM Model performance within LAMMPS
   - KIM Model pages
2. Installing the KIM API on Ubuntu (ppa:openkim/openkim)
3. Using LAMMPS with the KIM API

Appendix: Installing the KIM API by hand

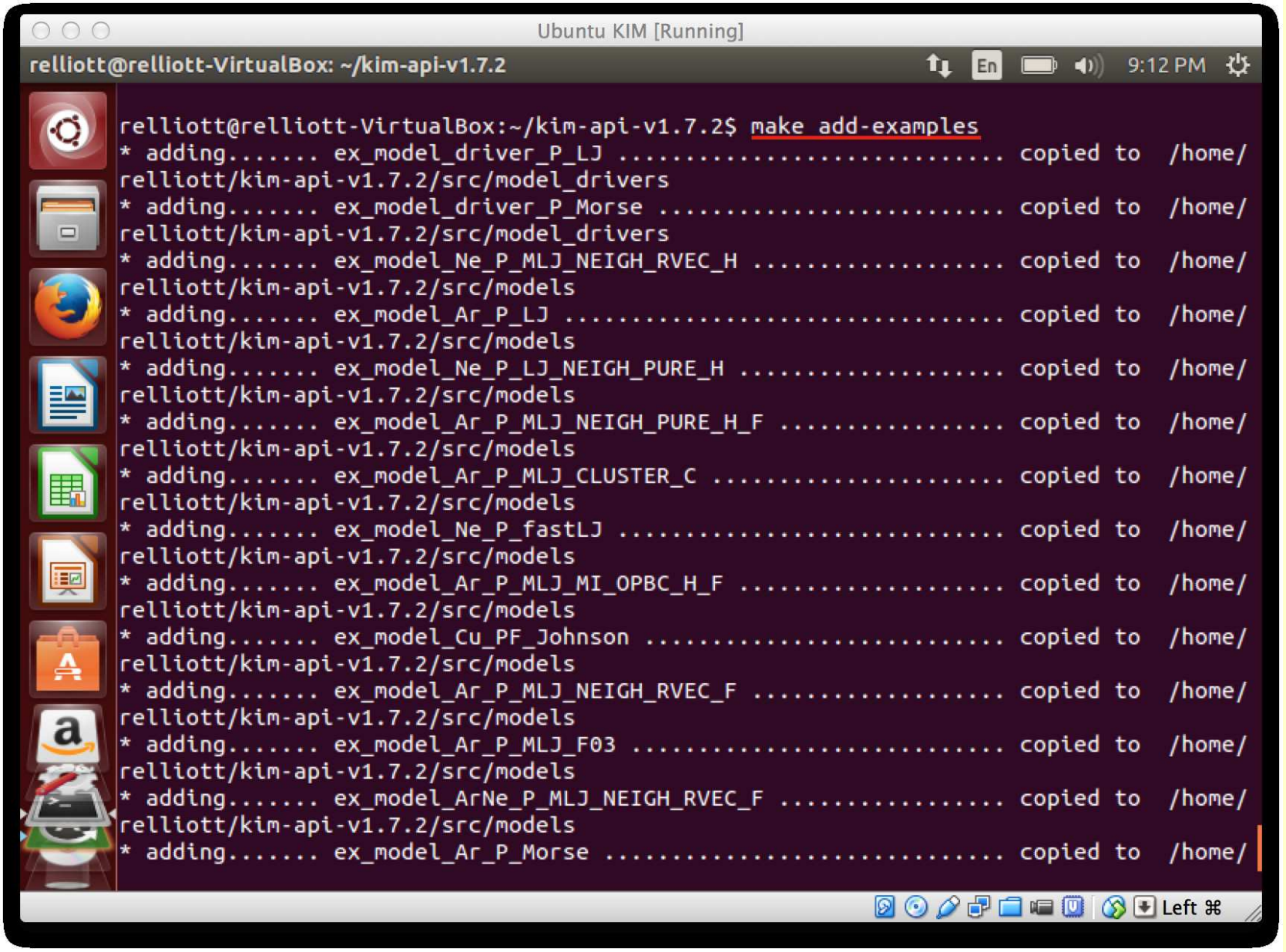
OpenKIM with LAMMPS



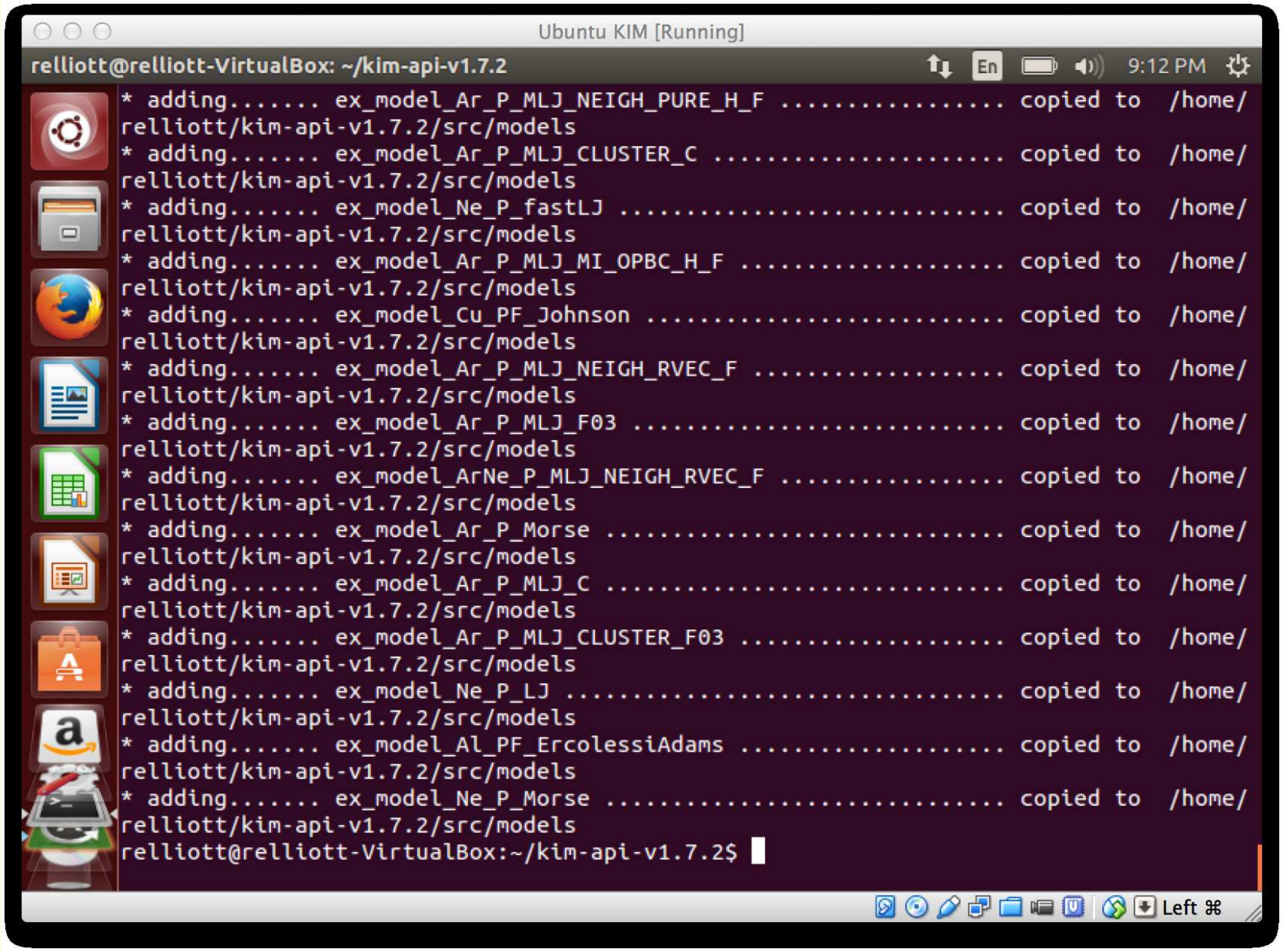
OpenKIM with LAMMPS



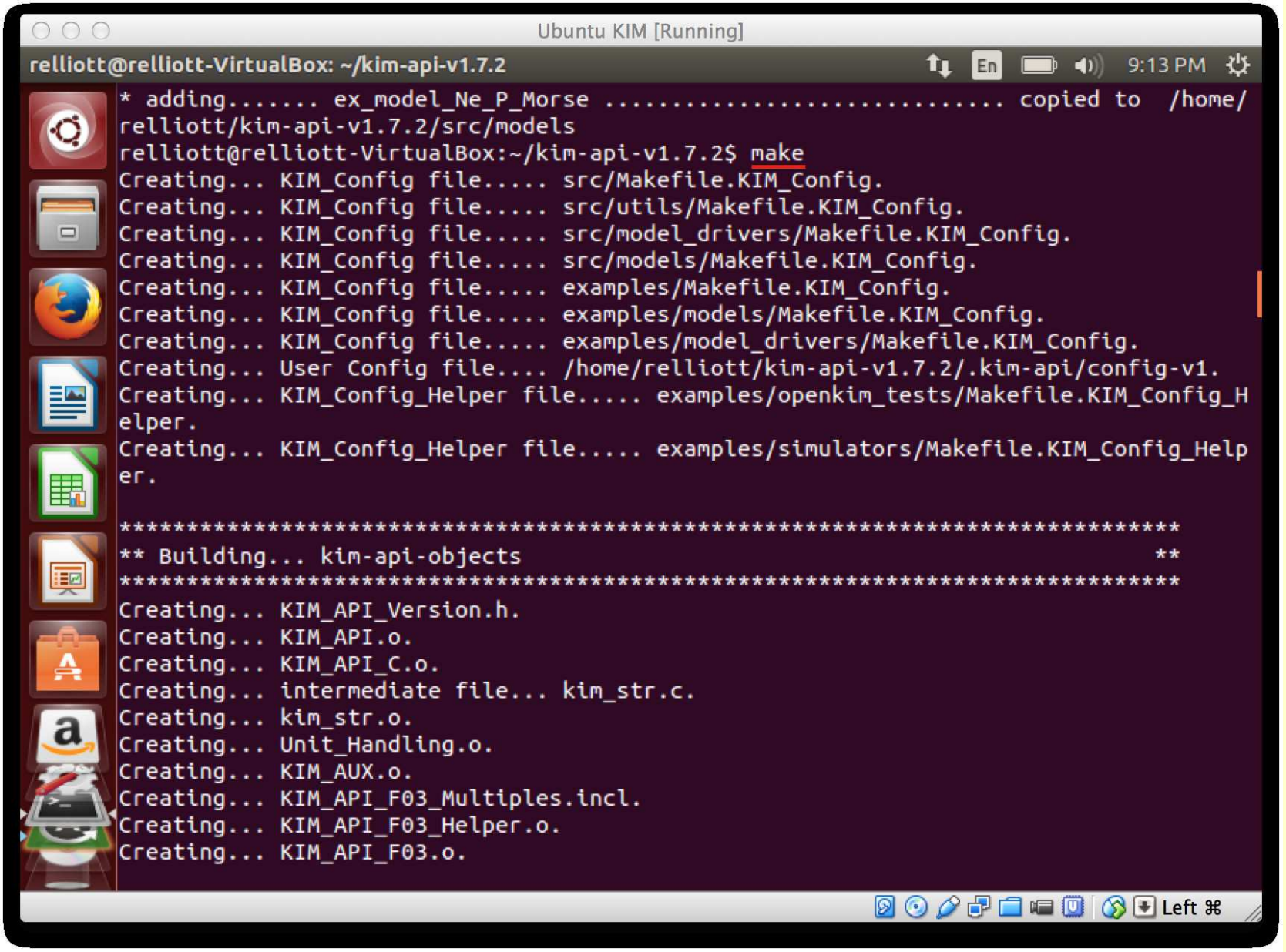
OpenKIM with LAMMPS



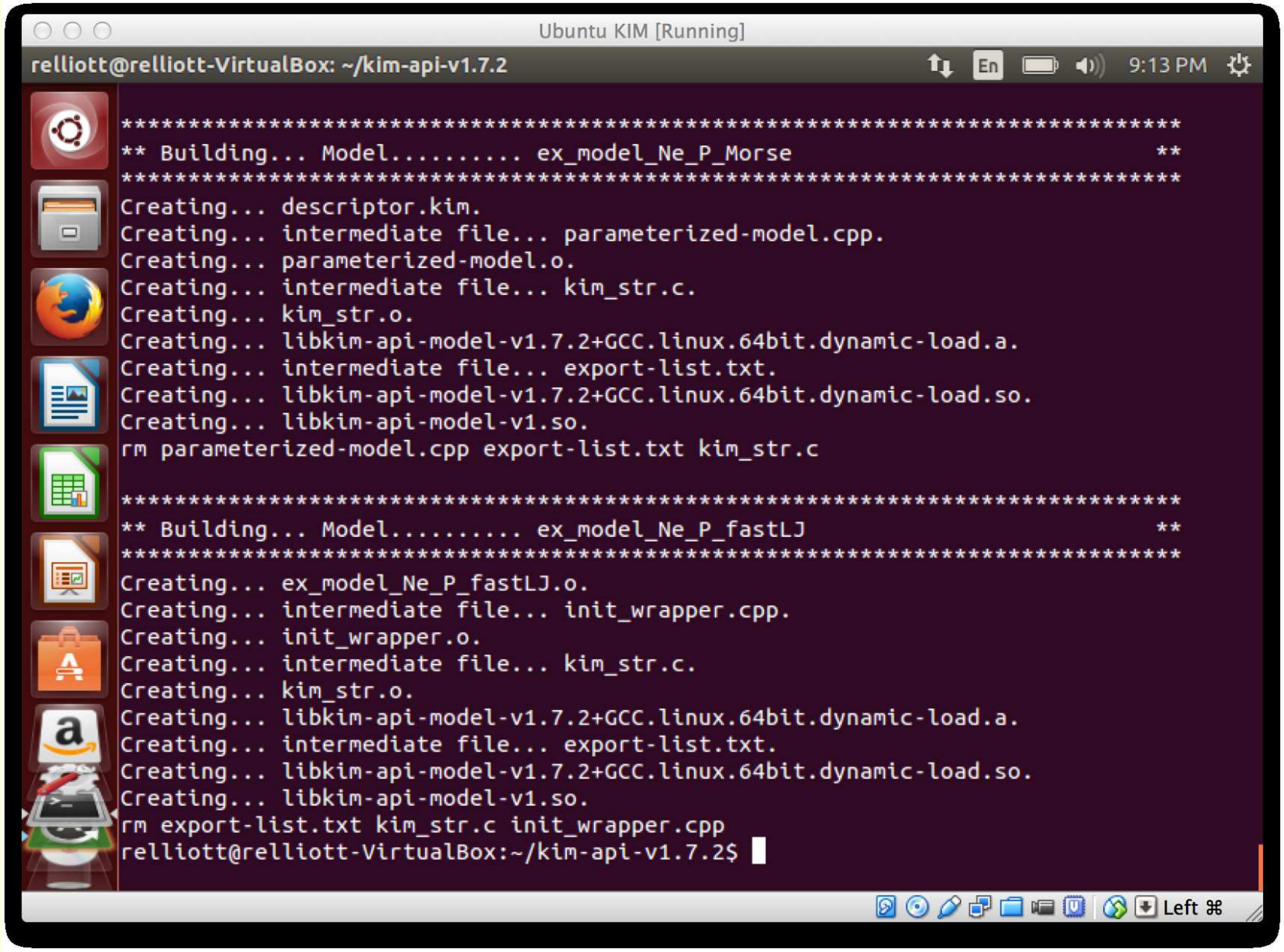
OpenKIM with LAMMPS



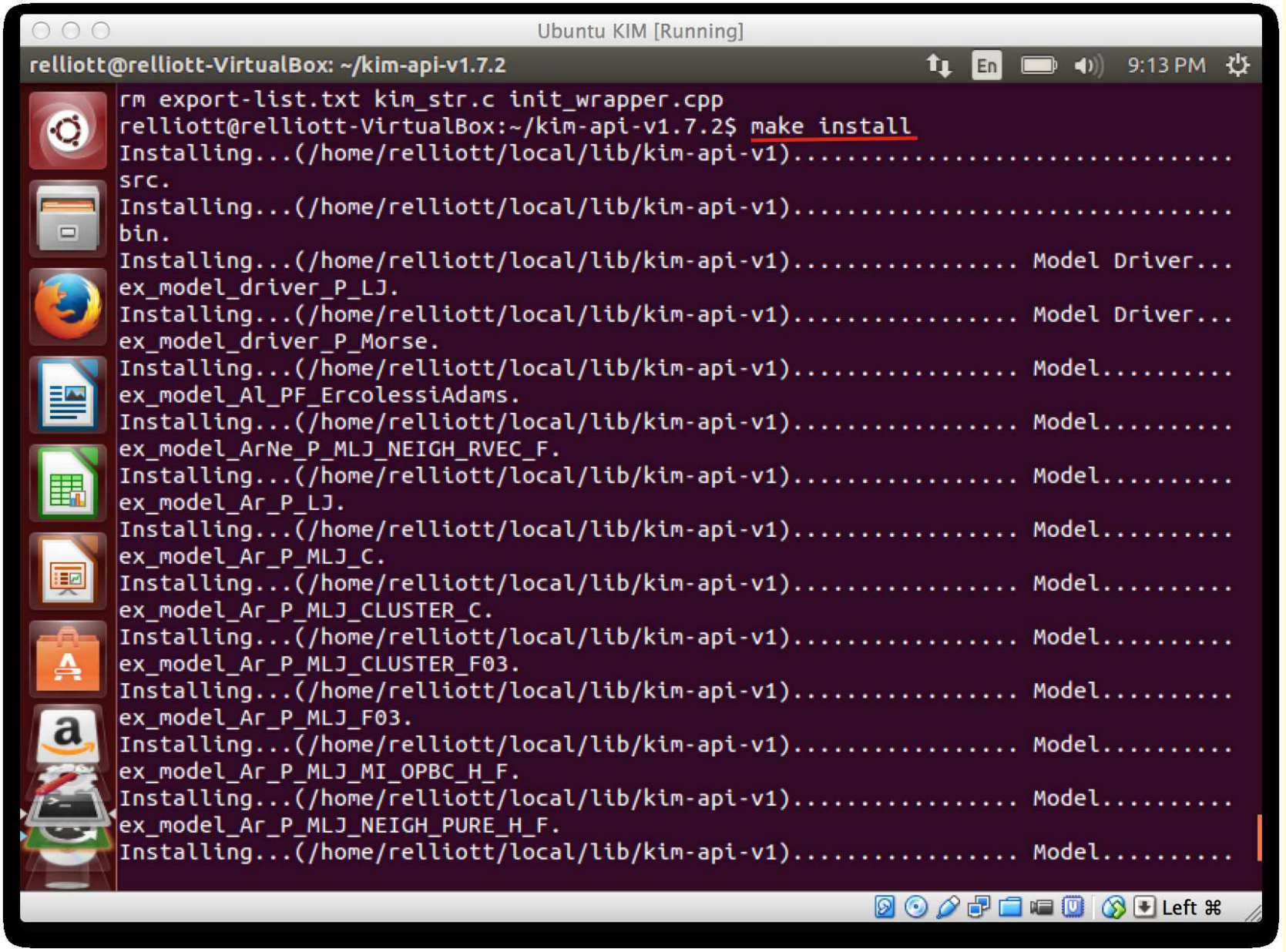
OpenKIM with LAMMPS



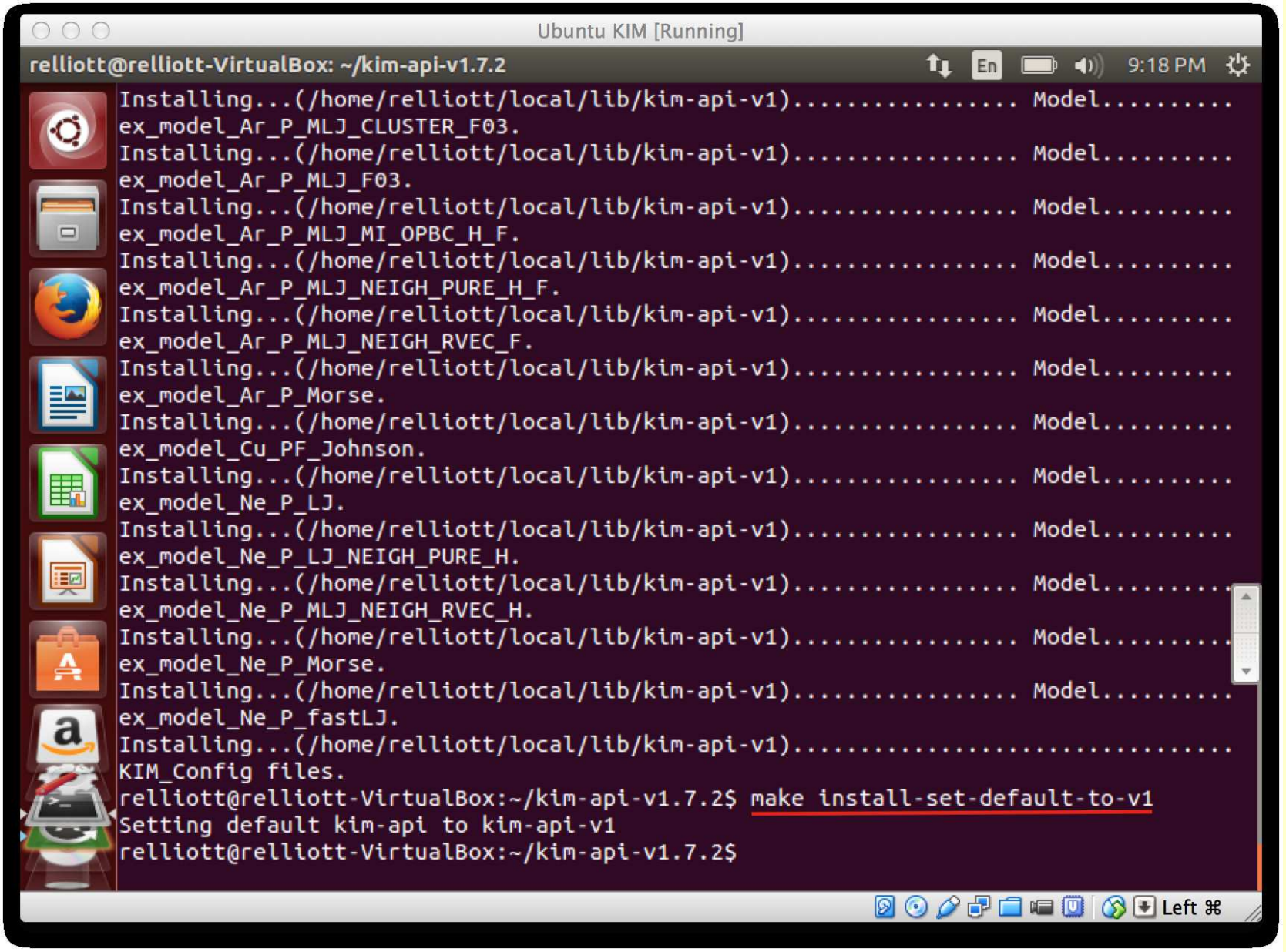
OpenKIM with LAMMPS



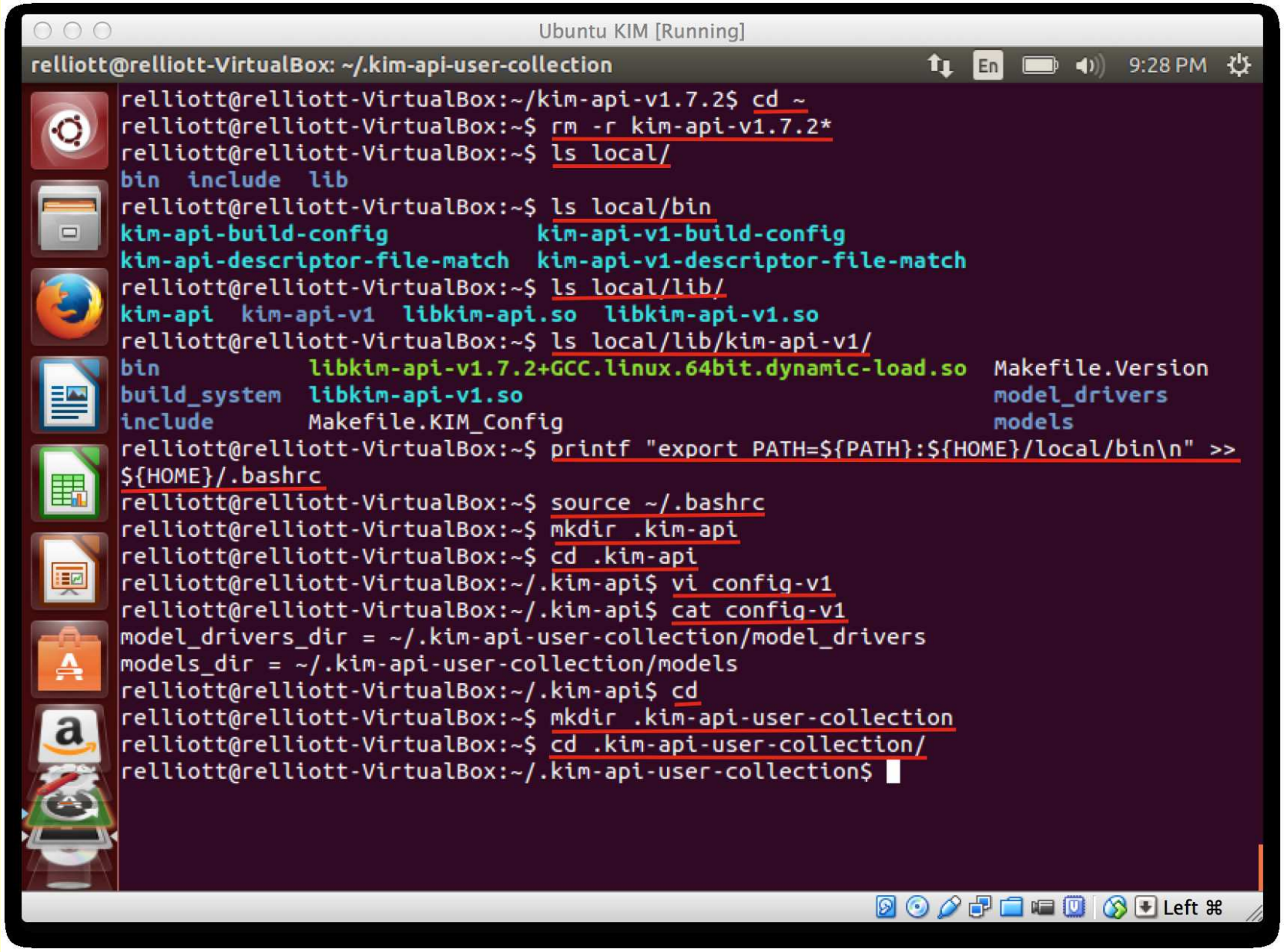
OpenKIM with LAMMPS



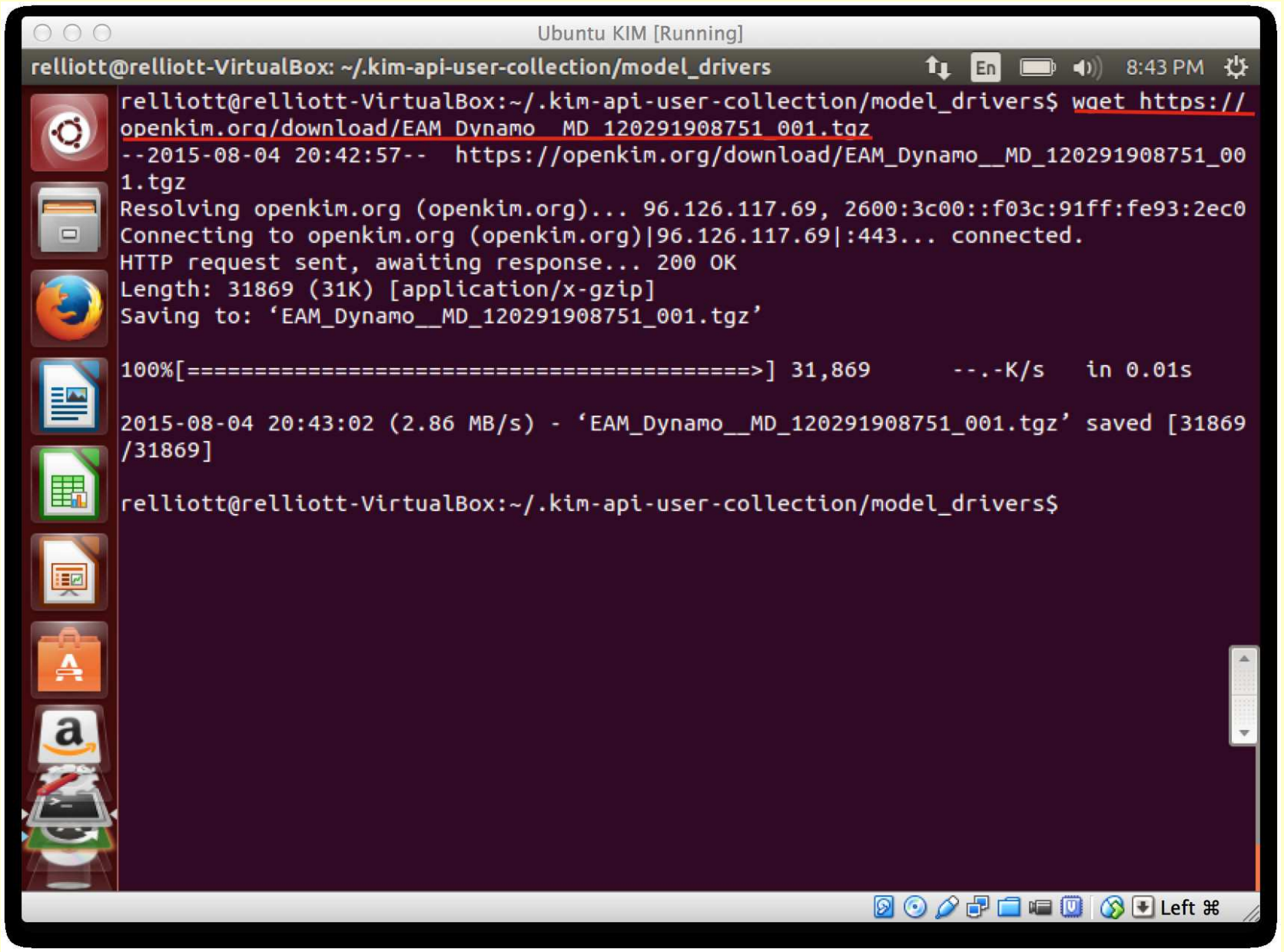
OpenKIM with LAMMPS



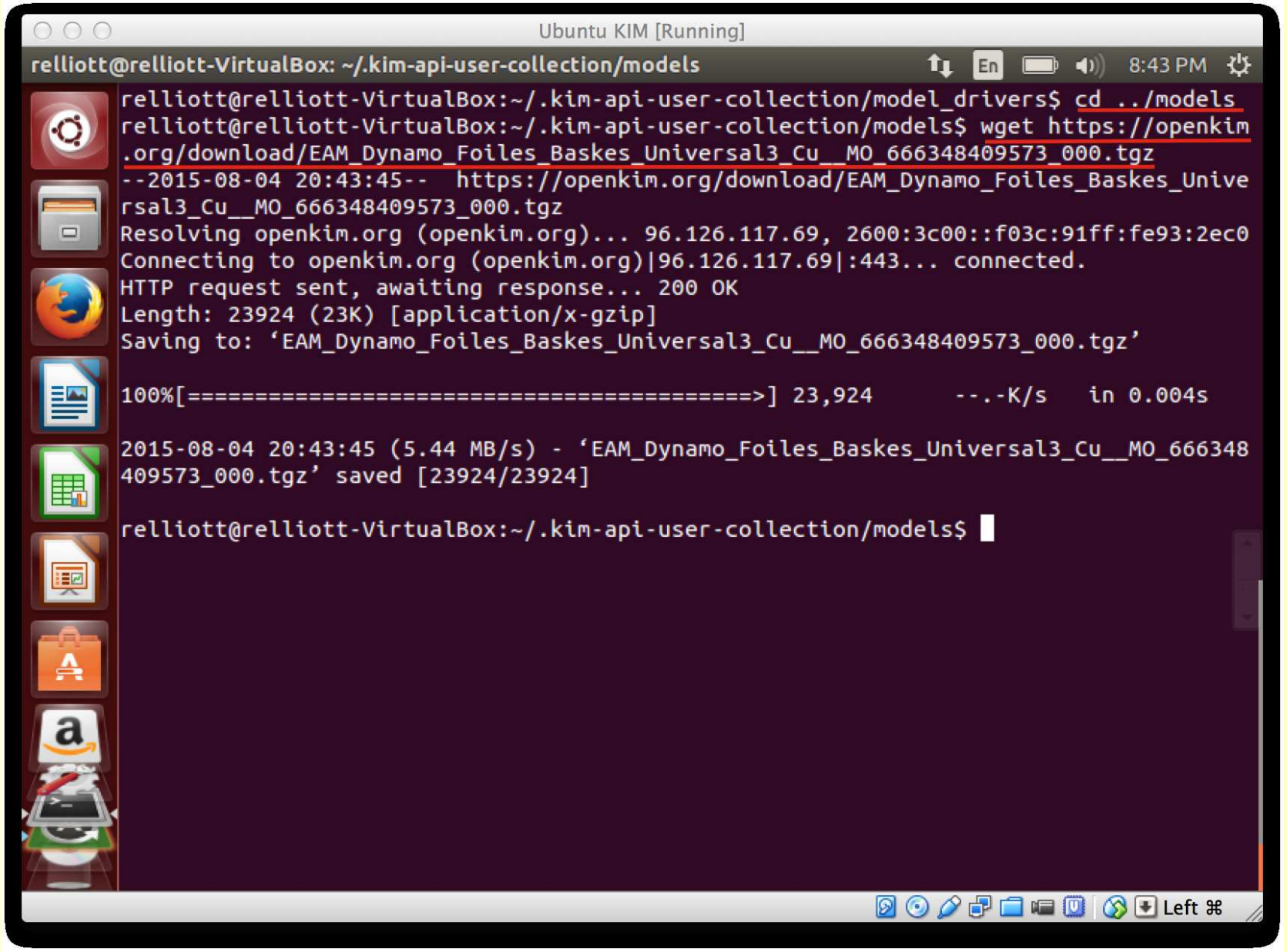
OpenKIM with LAMMPS



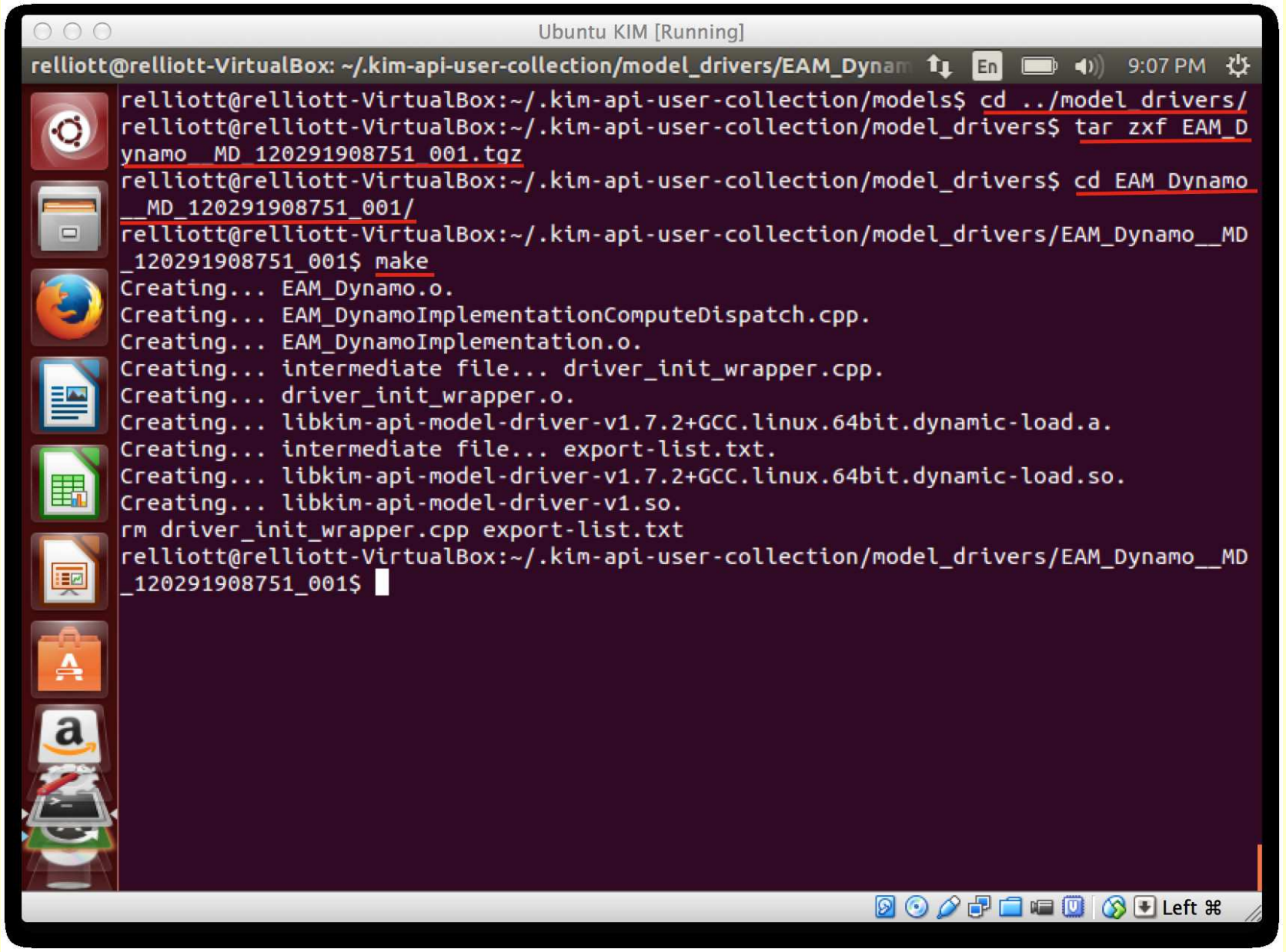
OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS



OpenKIM with LAMMPS