Computational nanoscience: LAMMPS tutorial

- What is LAMMPS?
  - Classical molecular dynamics code
  - Support for practically all most common potential models
    - Lennard-Jones, Morse, Tersoff, EAM, MEAM, ReaxFF, granular material, …
  - Support for many ensembles
  - Variable time step → usable in irradiation studies
  - Rich in features
  - Parallellized by MPI
  - Many potential models ported to GPGPU

- How to get the code
  - LAMMPS home page http://lammps.sandia.gov
    - Latest sources from the **svn** respository
      `svn co svn://svn.lammps.org/lammps-ro/trunk mylammps`
    - Or download the package from
      `http://www.acclab.helsinki.fi/~aakurone/download/mylammps.zip`
      and unpack it
      `unzip mylammps.zip`

Computational nanoscience: LAMMPS tutorial

- Commands for compiling LAMMPS

```
cd mylammps/src
make clean-all
make yes-manybody          Note1
make yes-MISC              Note1
make serial                Note2
```

- Notes
  - Note1: Thes are the LAMMPS packages needed in our simulations
  - Note2: This command does the compilation. It may take a while and you may see compiler warnings. If all went well you should see something like below on screen

```
size ../lmp_serial
   text    data     bss     dec     hex filename
3563874    7624     712 3572210  3681f2 ../lmp_serial
```

Computational nanoscience: LAMMPS tutorial

- The executable file produced is
  `mylammps/src/lmp_serial`
- You may test it by moving to folder `case1` in this package and giving command

```
/my/path/to/lammps/mylammps/src/lmp_serial -in in.case1
```


Computational nanoscience: LAMMPS tutorial

- What you need for this tutorial:
  - LAMMPS user manual in browser
    http://lammps.sandia.gov/doc/Manual.html
  - LAMMPS input files from package `lammps_starter_kit.tgz` on course Moodle page.
  - Open the package
    `tar xzvf lammps_starter_kit.tgz`
  - Read the `README` files

Computational nanoscience: LAMMPS tutorial

- LAMMPS input file
  - Define your simulation system **(A)**
    - Unit system
    - Particle properties
    - Interaction model
    - Boundary conditions, geometry (create lattice, read coordinates from file)
  - Define what is printed out **(B)**
    - `dump`: calculate and print per atom properties (potential and kinetic energies, atomic stresses,...)
    - `thermo`: print collective quantities (temperature, pressure, volume,...)
  - Script that tells what simulation runs are performed **(C)**
    - Which ensemble, desired temperature, desired pressure
    - How many timesteps
    - Many runs in the same input file
      - E.g. first heat the system, then cool down and finally optimize structure using the conjugate gradient algorithm

LAMMPS tutorial: LAMMPS input file

```
# NVE simulation of a cubic box of EAM copper
log             case1.log
```
Comments: number sign, output filename

```
units           metal
atom_style      atomic
```
Unit system, the kind of particles we have

(A)
```
boundary        p p p
lattice         fcc 3.61
region          mycube block 0 6 0 6 0 6
create_box      1 mycube
create_atoms    1 box
```
Build the system:
boundary conditions,
lattice structure,
shape of the system

```
mass            1 63.55
```

```
pair_style      eam
pair_coeff      1 1 Cu_u3.eam
neighbor        0.3 bin
neigh_modify    every 10 delay 0 check yes
```
Potential:
model,
parameter values,
neighbor list calculation

(B)
```
compute         ep all pe/atom
dump            mydump all custom 200 case1.dump &
                     type x y z c_ep
dump_modify     mydump append yes
```
Printing out of coordinates:
format,
what to print,
how often

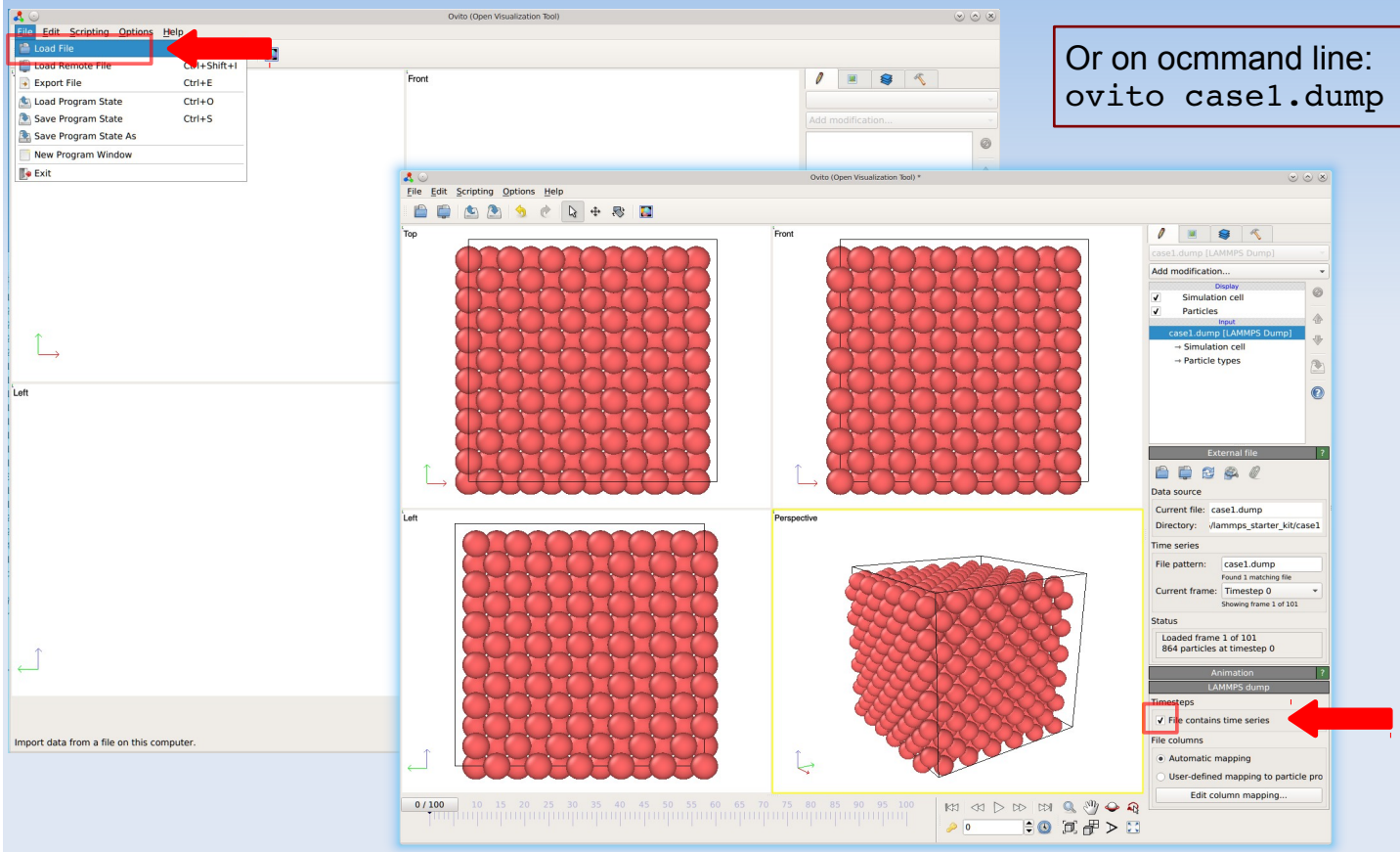
```
thermo          100
thermo_style    custom step temp pe etotal press vol
thermo_modify   line one flush yes format 1 "ec %8lu" &
                     format float "%20.10g"
```
Printing out of 'thermodynamical' data:
format,
what to print,
how often

(C)
```
fix             myfix all nve
velocity        all create 300.0 87287
run             5000
```
Setting related to simulation run:
ensemble ('fix'),
initial velocities,
run length

LAMMPS tutorial: Ovito

- Visualization program written for *materials scientists*
  - Not for biochemists like RasMol, JMol etc.
- Can read LAMMPS dump file formats directly
  - Animation (important!)
- Has some analysis capabilities (CNA, coordination, etc.)
- Home page: http://ovito.org
  - 'Easily' compiled from sources on Linux
  - Ubuntu package by Pekko Metsä
  - Binary packages for Linux, Windows and Mac.
    - Linux instructions:
      - Download the Linux binary package at http://ovito.org/index.php/download
      - Unpack it (the exact file name depends on the version; this is from 9.9.2015):
        `tar xzvf ovito-2.5.1-x86_64.tar.gz`
      - It you get error message '`gzip: stdin: not in gzip format`' leave the `z` out of the command , i.e.
        `tar xvf ovito-2.5.1-x86_64.tar.gz`
      - You can start Ovito by command
        `./ovito-2.5.1-x86_64/bin/ovito`

LAMMPS tutorial: Ovito

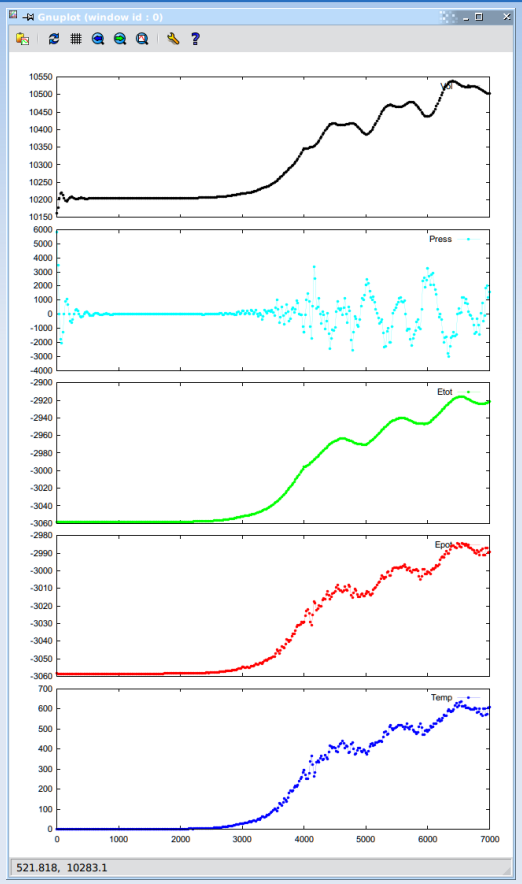


Or on ocmmand line:
`ovito case1.dump`

LAMMPS tutorial: Plotting LAMMPS data with gnuplot

- Directory `lammps_starter_kit/tools` contains a gnuplot script
- Edit the proper LAMMPS log file name to file `lmpplot.gp`
- Run by starting gnuplot and giving command

```
load './lmpplot.gp'
```



LAMMPS tutorial:

- If you have any questions ask the lecturer: antti.kuronen@helsinki.fi