

The screenshot displays the Atomify software interface. On the left is a sidebar with navigation options: run.in.min, simp..., betacristol, sica_sheet, New (#N), Open (#O), Close (#W), View, Analyse, Edit, Examples, Running, Stop, and Restart. The main window shows a simulation of atoms in a box, with a color scale on the right ranging from 0.00 (blue) to 62.22 (red). A central plot titled 'vsq' shows a distribution of values, with a peak at 12186.9. The right sidebar contains simulation and rendering controls, including a time slider (5.03), target speed (1x), and various data readouts for groups, regions, and computes. A footer note states: 'Atomify is developed by Anders Hafreager and Svernn-Arne Dragly, University of Oslo. Please report bugs here.'

INCREASED PRODUCTIVITY WITH REAL TIME VISUALIZATION

ATOMIFY

Anders Hafreager and Svernn-Arne Dragly
University of Oslo, Norway

**“I’M NOT GOING TO SHAVE
UNTIL I HAVE LEARNED
LAMMPS”**

76% male LAMMPS beginners [1]

[1] bit.ly/lammps-survey

**“I’M NOT GOING
UNTIL I HAVE
LAMMPS”**



76% male LAMMPS beginners [1]

[1] bit.ly/lammps-survey

DOWNLOAD FROM [BIT.LY/ATOMIFY](https://bit.ly/atomify)

LAMMPS WORKFLOW INCLUDES MULTIPLE PROGRAMS

```
in.vashishta.sio2
10 replicate      4 4 4
11 velocity      all create 2000.0 277387 mom yes
12 displace_atoms all move 0.05 0.9 0.4 units box
13
14 pair_style     vashishta
15 pair_coeff     * * Si0.1990.vashishta Si 0
16
17 neighbor      0.3 bin
18 neigh_modify  delay 10
19
20 fix           1 all nve
21 thermo       10
22 timestep     0.001
23
24 #dump         1 all cfg 10 *.cfg mass type xs ys zs vx vy vz fx fy fz
25 #dump_modify  1 element Si 0
26 dump         dump all atom 100 dump.atom
```

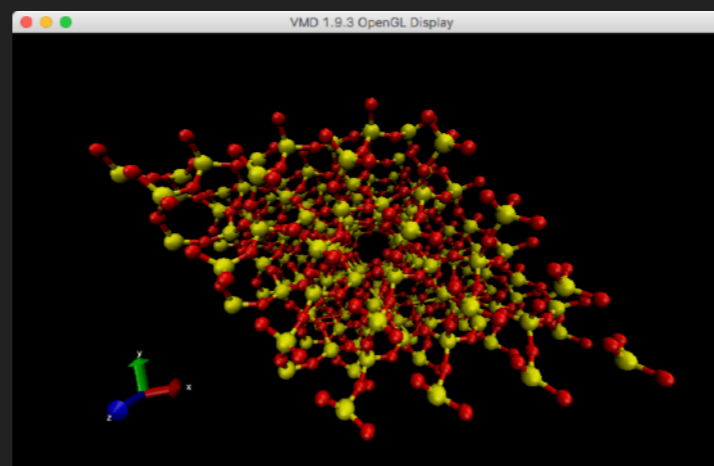
Editor to write scripts

```
1. fish /lammps/lammps/examples/vashishta (fish)
Time step      : 0.001
Per MPI rank memory allocation (min/avg/max) = 3.677 | 3.677 | 3.677 Mbytes
Step Temp E_pair E_mol TotEng Press
0          2000      -5280.875      0      -5132.2259      -20502.321
10        895.65237    -5198.402      0      -5131.833      419.34676
20        932.93463    -5201.1569     0      -5131.8169     -21407.961
30        936.09591    -5201.3998     0      -5131.8248     -32531.168
40        930.05159    -5201.0073     0      -5131.8816     -46445.212
50        904.64676    -5199.062      0      -5131.8245     -31402.385
60        1005.5353    -5206.5725     0      -5131.8365     -29790.442
70        941.02343    -5201.7644     0      -5131.8232     -23046.796
80        1020.1044    -5207.6763     0      -5131.8574     -13488.675
90        912.75535    -5199.651      0      -5131.8108     2715.5897
100       998.97588    -5206.1008     0      -5131.8523     6024.3651
Loop time of 1.50613 on 1 procs for 100 steps with 576 atoms

Performance: 5.737 ns/day, 4.184 hours/ns, 66.395 timesteps/s
99.1% CPU use with 1 MPI tasks x 1 OpenMP threads

MPI task timing breakdown:
Section | min time | avg time | max time | %varavg | %total
-----|-----|-----|-----|-----|-----
```

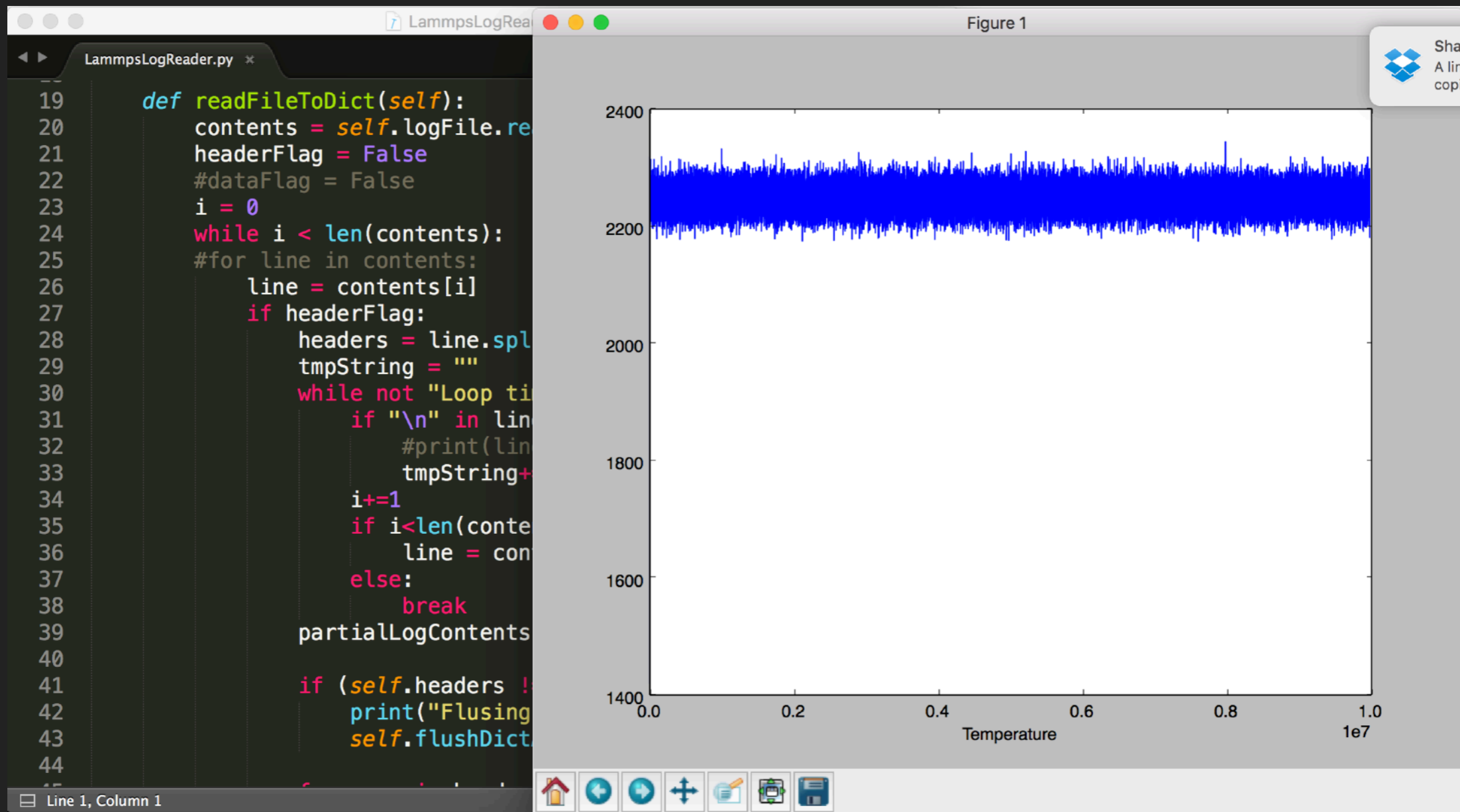
Terminal to run LAMMPS



VMD/Ovito to visualize

DOWNLOAD FROM [BIT.LY/ATOMIFY](https://bit.ly/atomify)

WHAT ABOUT PLOTTING?



ATOMIFY

- ▶ Uses LAMMPS as a library
- ▶ Real time visualization
- ▶ Supports accelerator packages
- ▶ Built with Qt (c++)
- ▶ Not trying to replace existing tools (VMD/Ovito)

The screenshot displays the Atomify software interface. On the left, a sidebar contains navigation options: View, Analyse, Edit, Examples, and a control panel with buttons for Stop and Restart. The main window is divided into several sections:

- Command-line Interface:** Shows a LAMMPS input script with variables for system size and region definitions. The script includes commands like `variable left equal 0.5*$L-0.5*$thickness`, `region system block 0 $L 0 $L 0 $L $L`, and `create_atoms 1 region boxinside`.
- 3D Visualization:** A central window displays a 3D visualization of a system of atoms, color-coded by temperature. A vertical color scale on the right ranges from 0.00 (blue) to 62.22 (red).
- Histogram Plot:** A window titled 'vsq' shows a histogram of the squared velocity of atoms. The x-axis ranges from 0.0 to 58.2, and the y-axis shows counts up to 12186.9.
- Simulation Control Panel:** Located on the right, it includes a 'SIMULATION' section with a progress bar (Time: 5.03, Target speed: 1x) and a 'RENDERING' section. It also lists 'Groups' (all: 16000 atoms, light: 2000 atoms, heavy: 14000 atoms) and 'Regions' (system: 15972 atoms, boxinside: 1560 atoms, boxoutside: 14440 atoms). A 'Computes' section lists variables like `thermo_temp: 2.782`, `thermo_press: 2.409`, and `msd_light`. A 'Variables' section lists parameters like `L: 20`, `Lhalf: 10`, and `thickness: 2`.

DOWNLOAD FROM [BIT.LY/ATOMIFY](https://bit.ly/atomify)

DEEMMO

WHAT CAN IT BE USED FOR?

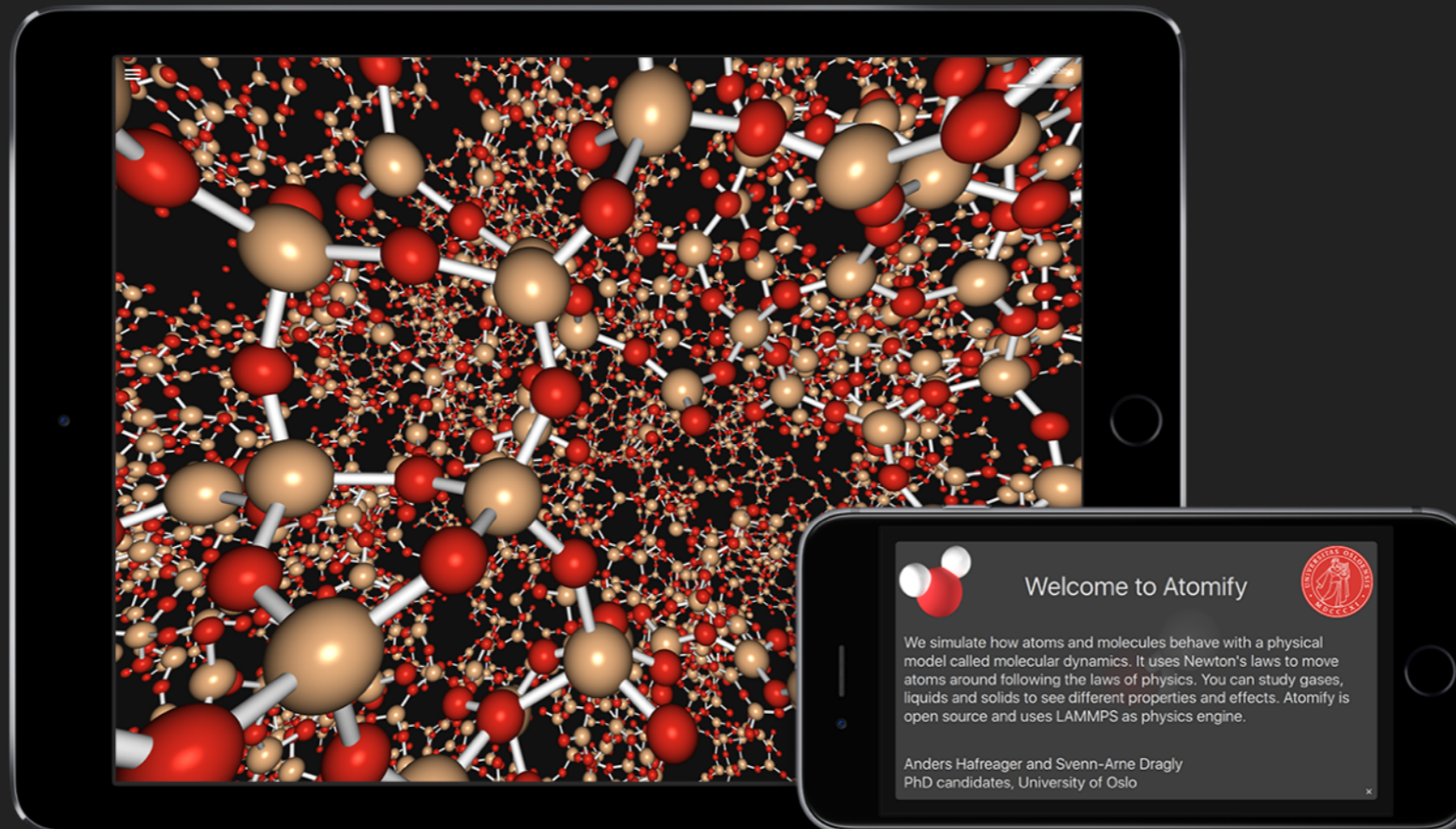
- ▶ Especially good for learning purposes
- ▶ Script prototyping
- ▶ With powerful desktop GPU's production runs can be performed
- ▶ I've run 10000 atoms, 1 microsecond with vashishta manybody potential (1xNVidia GP100, double precision)



SEARCH FOR ATOMIFY IN APP STORE

C++ COMPILES TO ALL PLATFORMS

- ▶ Run LAMMPS on your phone/tablet
- ▶ Free in App Store for both iOS and Android
- ▶ We need more systems! Do you have scripts that could be cool?



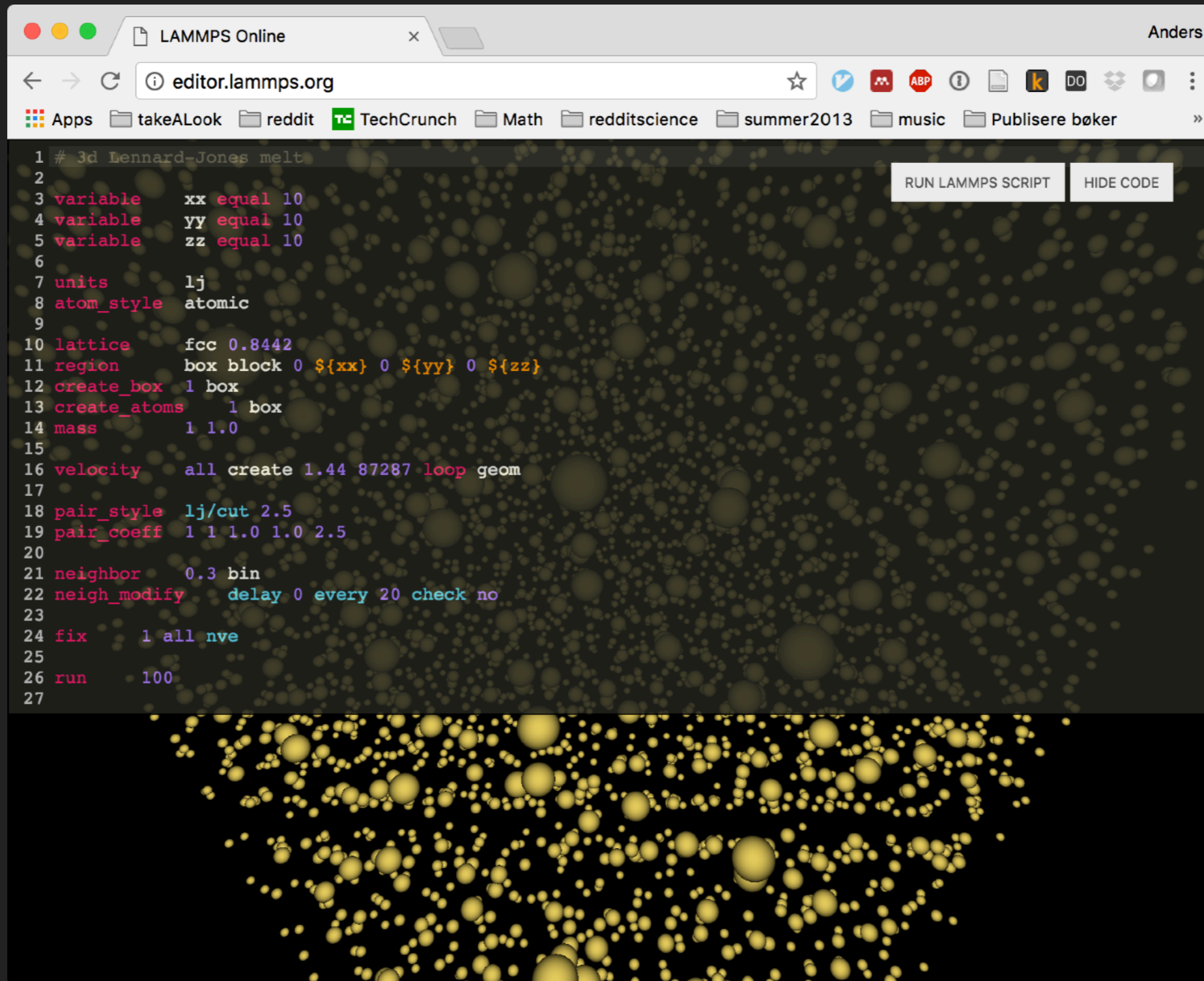
**AND BTW, HAS ANYONE HEARD ABOUT EMSCRIPTEN
AND ASM.JS/WEBASSEMBLY?**

C++ COMPILES TO JAVASCRIPT WITH EMSRIPTEN...



TRY IT AT EDITOR.LAMMPS.ORG

C++ COMPILES TO JAVASCRIPT WITH EMSRIPTEN...

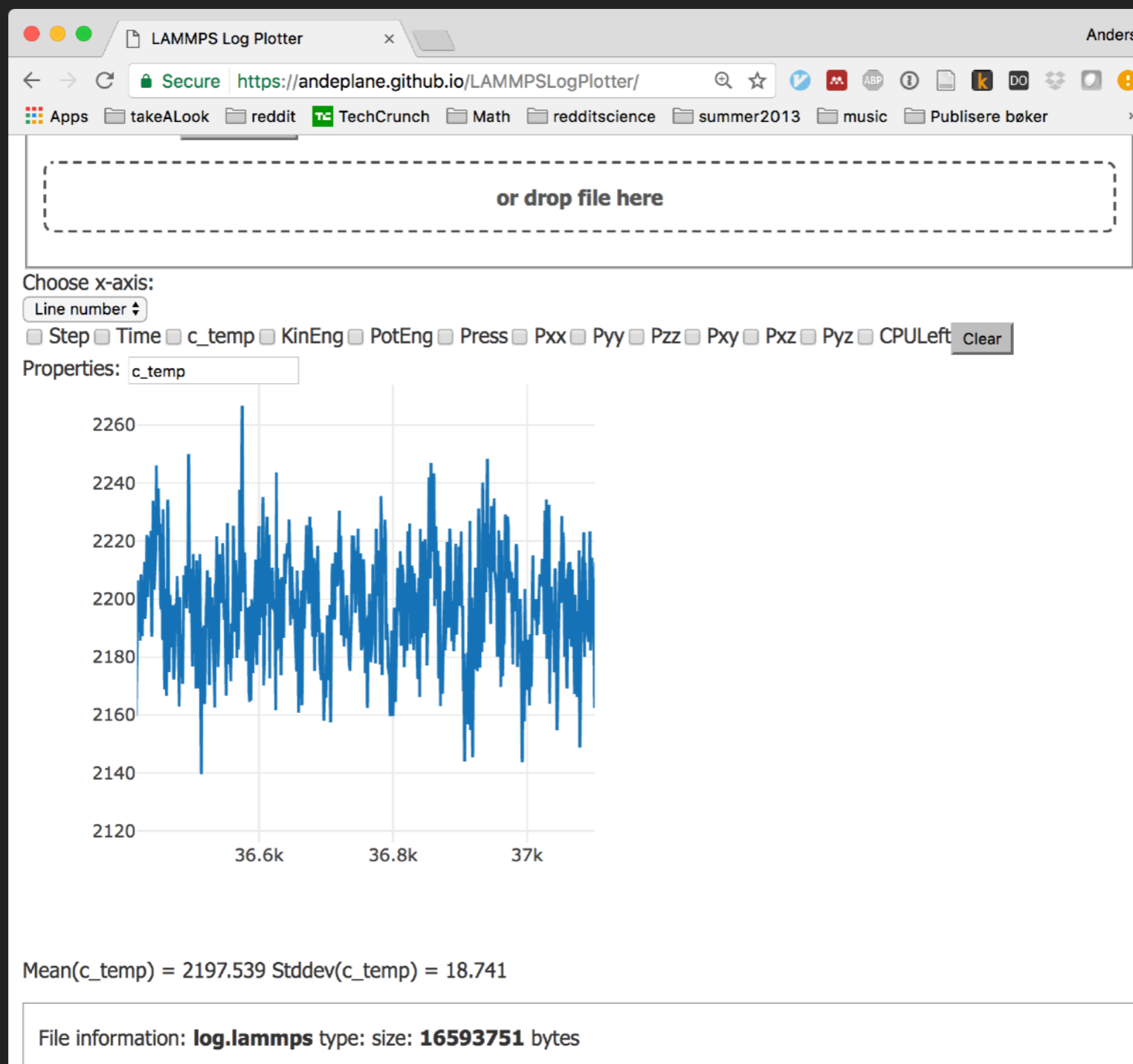


The screenshot shows a web browser window titled "LAMMPS Online" with the URL "editor.lammps.org". The browser's address bar and tabs are visible. Below the browser, there is a code editor with a dark background and light-colored text. The code is a LAMMPS script for a 3D melt simulation. The script includes commands for setting variables, units, lattice, region, create_box, create_atoms, mass, velocity, pair_style, pair_coeff, neighbor, neigh_modify, fix, and run. The script is numbered from 1 to 27. To the right of the code editor, there are two buttons: "RUN LAMMPS SCRIPT" and "HIDE CODE". Below the code editor, there is a 3D visualization of a melt simulation, showing a dense collection of yellow and orange spheres representing atoms in a disordered state.

```
1 # 3d Lennard-Jones melt
2
3 variable      xx equal 10
4 variable      yy equal 10
5 variable      zz equal 10
6
7 units         lj
8 atom_style    atomic
9
10 lattice      fcc 0.8442
11 region       box block 0 ${xx} 0 ${yy} 0 ${zz}
12 create_box   1 box
13 create_atoms 1 box
14 mass         1 1.0
15
16 velocity     all create 1.44 87287 loop geom
17
18 pair_style    lj/cut 2.5
19 pair_coeff    1 1 1.0 1.0 2.5
20
21 neighbor     0.3 bin
22 neigh_modify delay 0 every 20 check no
23
24 fix          1 all nve
25
26 run          100
27
```

TRY IT AT [BIT.LY/LOGPLOTTER](https://bit.ly/logplotter)

EASY PLOTTING OF LOG FILES



The screenshot shows a web browser window titled "LAMMPS Log Plotter" with the URL <https://andepiane.github.io/LAMMPSLogPlotter/>. The browser's address bar shows "Secure" and the URL. The page features a dashed box with the text "or drop file here". Below this, there is a section for "Choose x-axis:" with a dropdown menu set to "Line number" and a row of checkboxes for various properties: Step, Time, c_temp, KinEng, PotEng, Press, Pxx, Pyy, Pzz, Pxy, Pxz, Pyz, and CPULeft. A "Clear" button is located to the right of these checkboxes. The "Properties:" field is set to "c_temp". The main plot area displays a line graph of "c_temp" over time. The y-axis ranges from 2120 to 2260, and the x-axis ranges from 36.6k to 37k. The plot shows a highly oscillatory blue line. Below the plot, the statistics are displayed: Mean(c_temp) = 2197.539 and Stddev(c_temp) = 18.741. At the bottom, the file information is shown: File information: **log.lammps** type: size: **16593751** bytes.

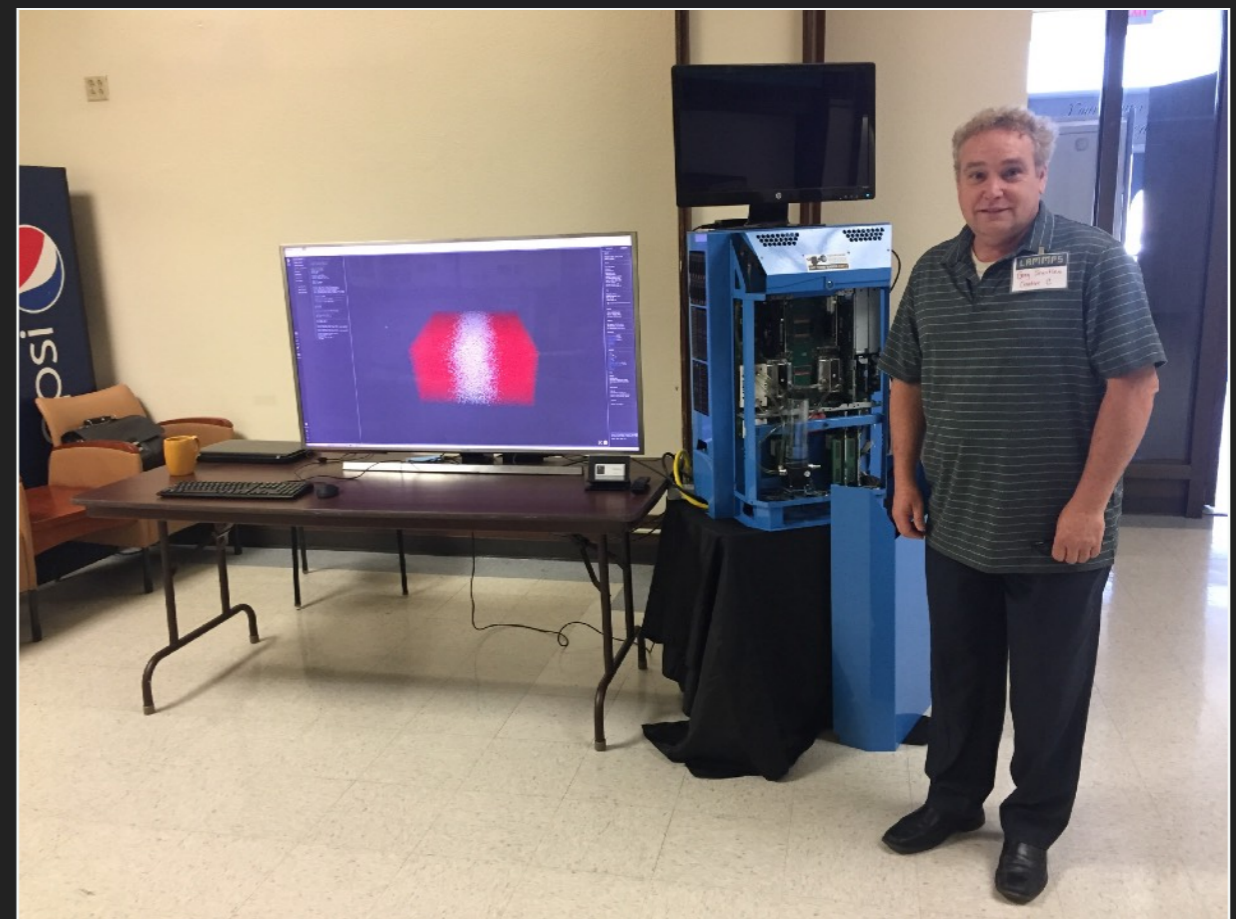
DOWNLOAD FROM [BIT.LY/ATOMIFY](https://bit.ly/atomify)

IF YOU WANT TO TRY

- ▶ Phone/tablet version of Atomify is free in iOS / Android app stores
- ▶ Download Atomify for macOS or Ubuntu:
github.com/ovilab/atomify
or bit.ly/atomify
- ▶ Web version: editor.lammps.org
- ▶ Log plotter: bit.ly/logplotter

▶ **Feedback is extremely appreciated!**

Thank you!
(sorry, I had to)



Check out the demo during lunch!
(thanks Greg)