

Table of Contents

Table of Contents	1
LAMMPS trajectories into QuantumATK	2
Preparing a LAMMPS script	2
Running LAMMPS	3
Importing to QuantumATK	3
Postprocessing LAMMPS trajectories	6
References	8



LAMMPS trajectories into QuantumATK

Category: QuantumATK as GUI for LAMMPS

Links: [LAMMPS website](#), [Notes on LAMMPS](#)

[PDF version](#)

With QuantumATK you can import, visualize, and analyze output data generated by a variety of other simulation codes like VASP, Quantum ESPRESSO, and many more. It is also possible to import classical MD trajectories generated by LAMMPS, and use for example the Movie Tool, the 3D Viewer, or the MD Analyzer to take a closer look at your simulations.



LAMMPS is a classical molecular dynamics code ^[1] with a very flexible output format for its trajectories. Most of the available output is written as text files. In order to import LAMMPS trajectories into QuantumATK and make full use of the QuantumATK analysis functionalities, you have to **stick to some minimal rules when running LAMMPS simulations**, which are explained in this tutorial.

You will learn how to import your LAMMPS simulation trajectories into QuantumATK and make use of the visualization and analysis tools, and how to prepare input files for further simulations based on the imported configurations.

Preparing a LAMMPS script

A LAMMPS simulation is typically controlled by an input script, in which all the simulation parameters are listed. For details about the LAMMPS commands and usages, see the [LAMMPS Documentation](#). The output trajectory is specified by the **dump** bash command. QuantumATK can currently read trajectory files written in the **atom** or **custom** style. The format of the atom style is fixed, so no further specifications have to be made, and the command may simply look like this:

```
dump          d_trajectory all atom 100 trajectory.dat
```

It will write the coordinates of all atoms every 100 steps to the file `trajectory.dat`. The extension of the trajectory file does not need to be a specific one for QuantumATK, as long as it does not conflict with other extensions. The atom style does not write velocities, so the resulting trajectories can not be imported as `MDTrajectory` objects (as required for the **Movie Tool** or the **MDAnalyzer**), but only as regular

Trajectory objects. Nevertheless, they can be visualized by the **3D Viewer** and drag-and-dropped onto the **Builder**.

Tip

The LAMMPS custom format allows you to specify explicitly which quantities should be recorded. For compatibility with QuantumATK, at least *type*, *index*, and the *coordinates* (x, y, z, or xs, ys, zs) have to be present in the trajectory file. To be imported as an **MDTrajectory** object, the trajectory file also needs to have velocities. This may be accomplished by this command:

```
dump          d_trajectory all custom 100 trajectory.dat id type x y z vx vy vz
```

For **MDTrajectory** objects you also need to extract the potential energies. These can be read from the thermodynamic output in the log file that is created during the simulation. Therefore, the thermodynamic output from LAMMPS must be written with the same frequency as the trajectory. You can specify the frequency via the keyword *thermo*. Moreover, you have to make sure that the potential energies are present in the thermodynamic output, which can be selected under the *thermo_style* keyword. A typical example (compatible with one of the above trajectory dump commands) might look like this:

```
thermo        100
thermo_style  custom step temp pe etotal press
```

Here, the *pe* flag specifies that potential energies should be written to the log file. If all these requirements are fulfilled, the trajectory is read by QuantumATK as an **MDTrajectory**. Alternatively, if only the minimal requirements are fulfilled, QuantumATK will read it as a regular **Trajectory** object.

Running LAMMPS

Since LAMMPS provides a well defined output format and initial configuration, QuantumATK simply reads at first the log file, in which all these settings are mirrored during a LAMMPS simulation. Based on the statements in this log file, the output files and other possible input files are read. Therefore, you need to make sure that the LAMMPS log file is written properly. You can specify the log file output via the option *-log* in the LAMMPS command line:

```
lmp_serial -in in.settings -log log_simulation_1.lammps
```

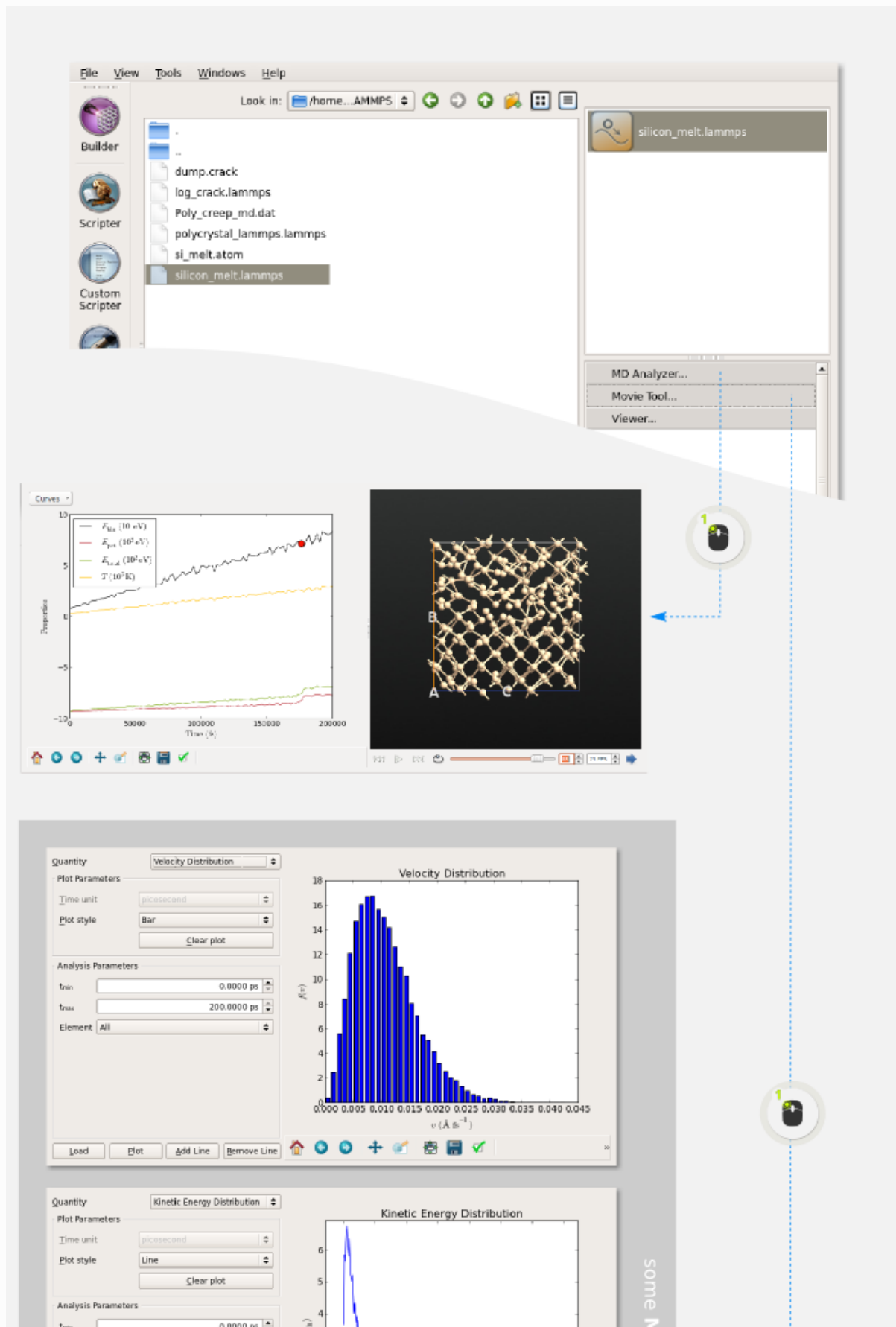
Attention

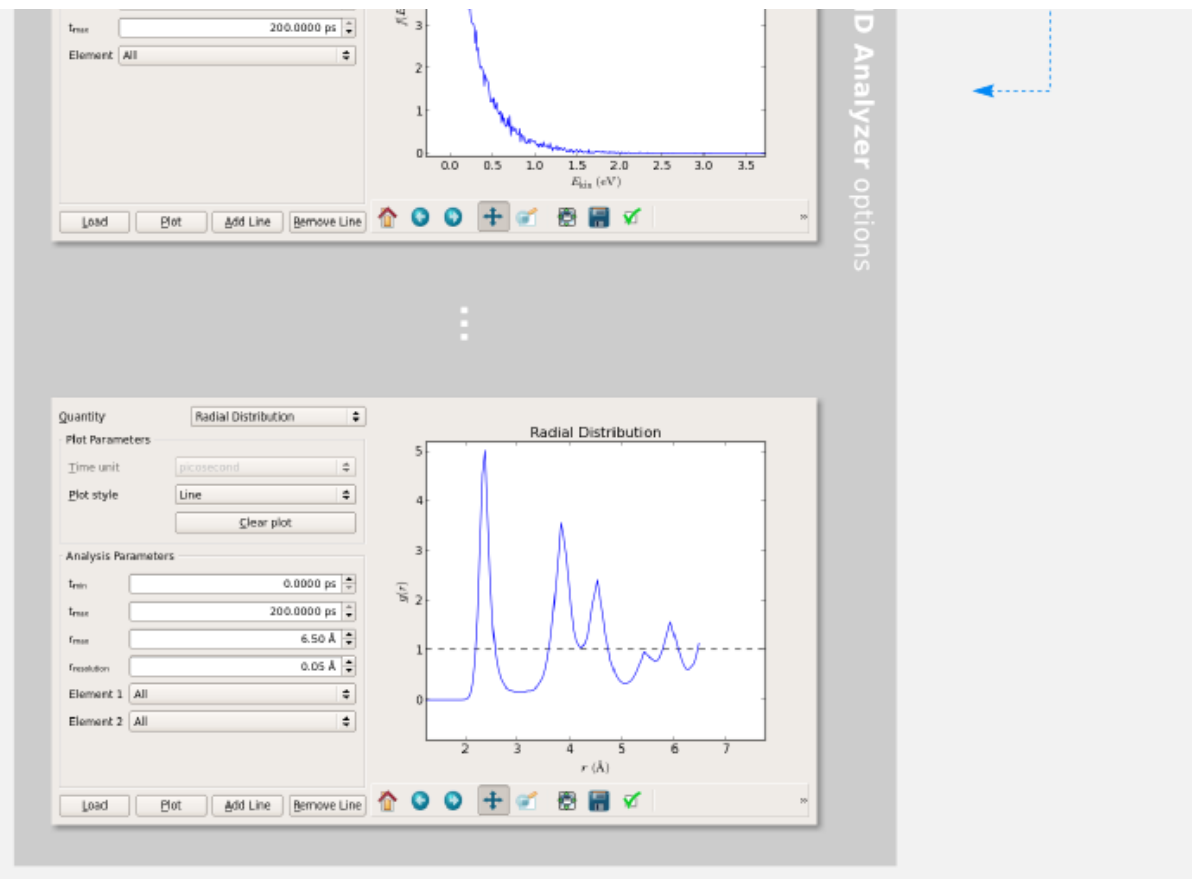
You have to select a file name ending with `.lammps`, as this extension is recognized by QuantumATK as a LAMMPS log file. If you are reading your input configuration from a separate file (typically starting with `data.*`) you also need to make sure that this file is present in your current directory, since QuantumATK reads the atomic masses from it.

Importing to QuantumATK

If you have followed the guidelines in the previous section, the trajectory should show up on the **LabFloor** in QuantumATK after the LAMMPS simulation. It will be labelled according to the name of the log file without extension, i.e. the example simulation of the previous section will be named `log_simulation_1`. If you click the icon, you will see which plugins are available for this particular trajectory on the right-hand side panel. For an **MDTrajectory** object you can use the **Viewer**, the **Movie Tool**, and the **MDAnalyzer**. For a regular **Trajectory** you can only use the **Viewer**.

In the following example, a silicon crystal has been melted by linearly increasing the temperature from 300 to 3000 K over 200 ps at zero pressure. The Stillinger-Weber potential has been used to describe the interactions among the silicon atoms. The resulting LAMMPS trajectory shows up on the LabFloor and can be visualized using the Movie Tool.





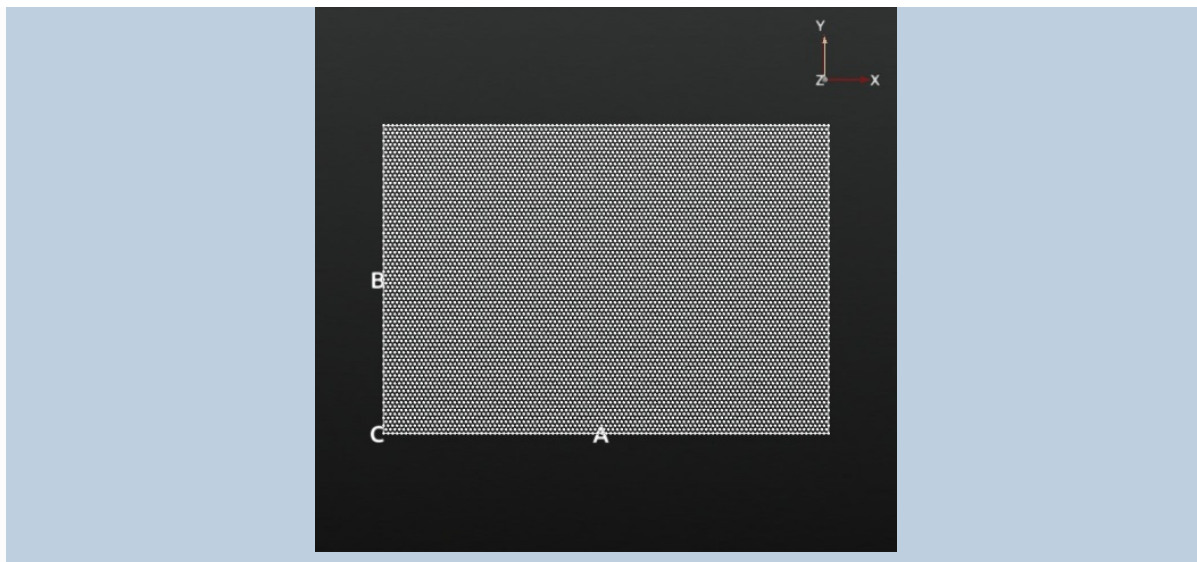
Note

Dimensionless units

LAMMPS simulations are sometimes performed using a system of dimensionless units, enabled by the keyword **units lj**, together with generic atom types. You can of course import such trajectories into QuantumATK.



However, as there are no physical units associated with the velocities and energies, the trajectory will be imported as a regular **Trajectory** object. The example below shows the initiation and propagation of a crack in a 2-dimensional system of generic Lennard-Jones atoms.


If you want the full functionality of an **MDTrajectory** you should preferably use either **units metal** (eV, Ang, ps) or **units real** (kcal/mol, Ang, fs) in your LAMMPS simulation.




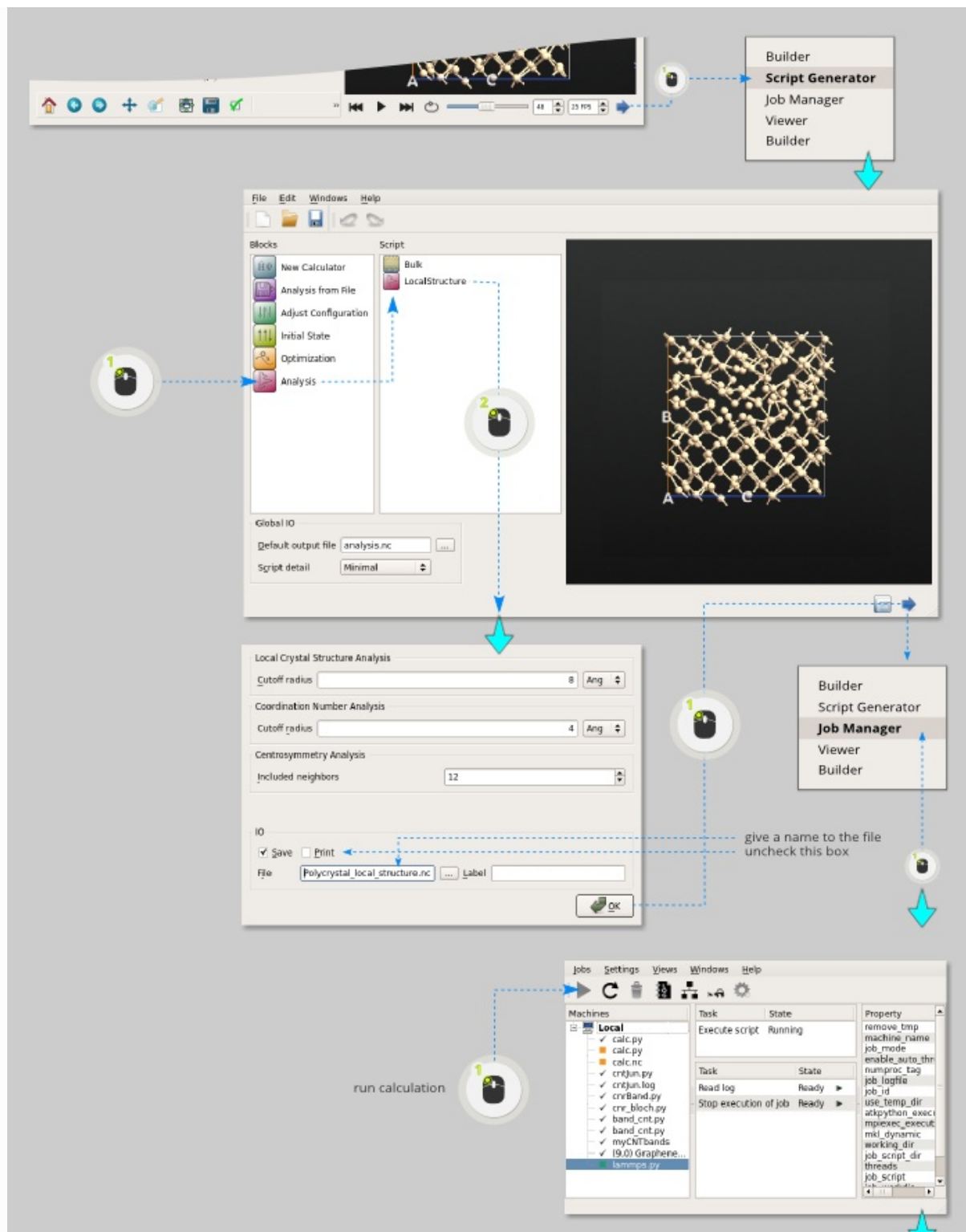
Postprocessing LAMMPS trajectories

If you have imported a LAMMPS trajectory to the QuantumATK LabFloor, you can of course extract single configurations, and perform further analyses or calculations using QuantumATK. The following example shows how you can calculate the local structure of a copper polycrystal that has been simulated with LAMMPS. If you want to learn how to generate and simulate polycrystals in QuantumATK, take a look at the tutorial [Simulating a creep experiment of polycrystalline copper](#).

Open the LAMMPS trajectory with the Movie Tool. Go to the desired snapshot, e.g. the last snapshot, and send this configuration to the Script Generator  using the  icon in the lower right-hand corner of the Movie Tool widget.

In the Script Generator, add an  Analysis ► LocalStructure block. Open the block by double-clicking it, select a suitable file name, e.g. `Polycrystal_local_structure.nc`, uncheck the *Print* box (the structure is huge and will otherwise produce a large amount of text output), and click *OK* to close the LocalStructure settings.

Run the calculation using the Job Manager .



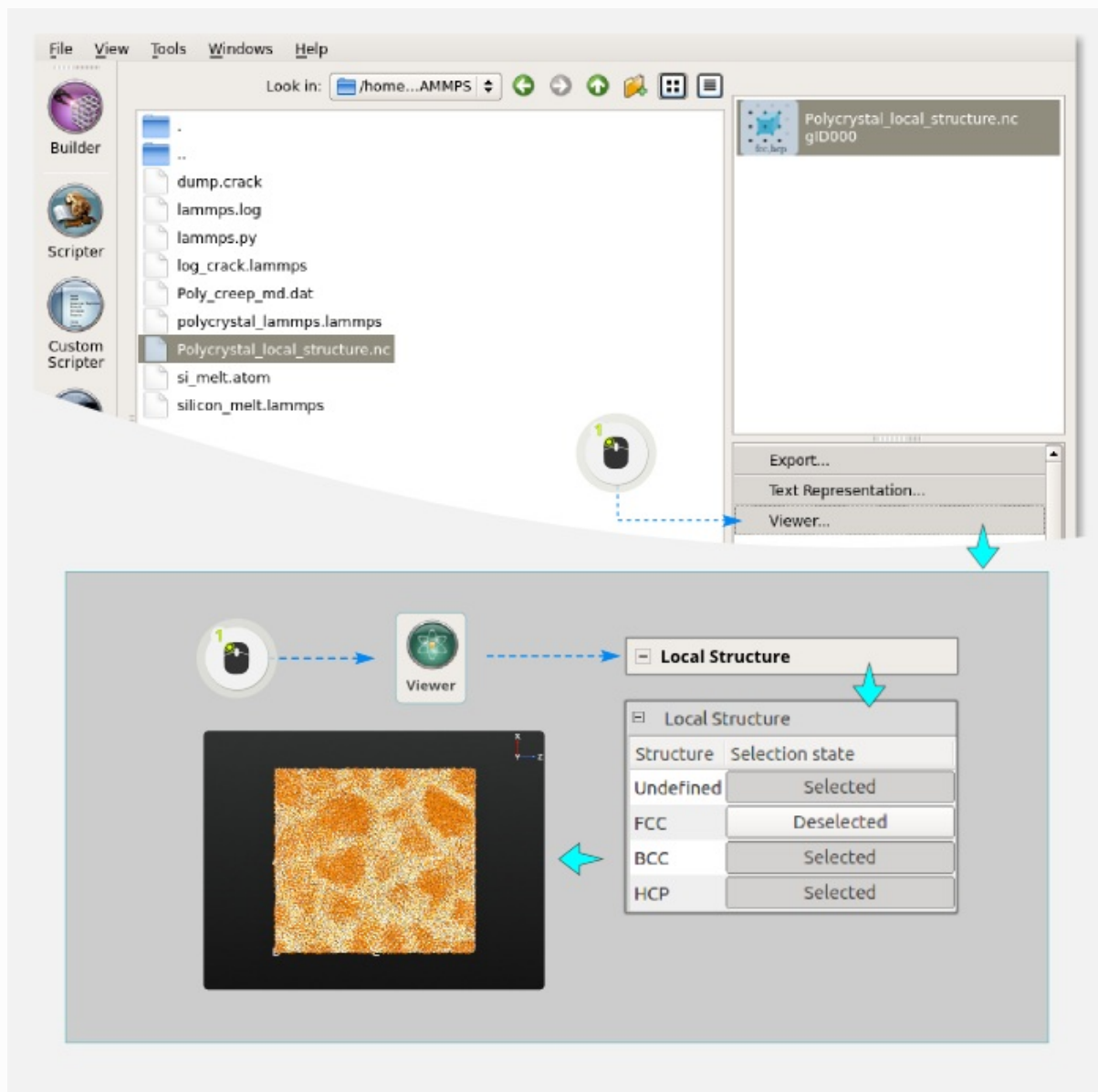
After the calculation has finished, the local structure file will appear on the LabFloor.




LocalStructure

Open it in the Viewer and find the Local Structure plugin in the panel bar of the Viewer. Expand it and you will see a list of local crystal structures. In this example we will highlight the grain boundaries of the polycrystal:

- Select all structure types except FCC, which is the native crystal structure inside the silicon grains.
- Click the **Properties** plugin to open the scene properties. In the *Atoms* tab, change the color of the selected atoms, e.g. to white, to distinguish these atoms from grain atoms.
- Additionally, you can click on *Ball*, to switch off the bonds within the grain boundaries, which increases the transparency of the polycrystal.
- When you close the Properties widget, your changes become active and you will find the grain boundaries clearly visible.



Besides local structure analysis, you can also perform further simulations, e.g. using the **ATK-DFT** engine for density functional theory calculations on top of the extracted snapshots: Simply add a  **New Calculator** block in the Script Generator. This provides an easy way to interface your classical LAMMPS simulations with quantum mechanical calculations.

References

[1]

Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1 – 19, 1995. doi:10.1006/jcph.1995.1039.

© Copyright 2023 Synopsys, Inc. All Rights Reserved.