Table of Contents

Pymatgen in QuantumATK

Version: 2015.1

Downloads & Links

⬇ PDF version
Pymatgen
Materials Project

Pymatgen (Python Materials Genomics) is an open-source Python library for materials analysis. It currently powers the public Materials Project, an initiative to make calculated properties of all known inorganic materials available to materials researchers.

QuantumATK 2016 and later ships with pymatgen, while users of earlier versions will need to manually install it. This tutorial therefore explains how to install and update the pymatgen package for use with atkpython.

> **❶ Note**
>
> Installing and updating pymatgen requires that a working **c** compiler is available on the system. There should in principle be no issues using a different compiler than the one atkpython was compiled with, although it is not guarenteed to work.

Installation

Pymatgen can be installed using Python's *easy install* mechanism, which will ensure that all dependencies are handled:

```
atkpython -measy_install pymatgen
```

This will download and install pymatgen. If you are installing on a system without access to the internet, it is possible to install directly from a package:

```
atkpython -measy_install PACKAGE_PATH
```

The newest package is available from https://pypi.python.org/pypi/pymatgen.

Testing

To test that the installation of pymatgen was successful, run `atkpython` and then type the following commands at the prompt:

```
>>> import pymatgen as mg
>>> si = mg.Element("Si")
>>> si.atomic_mass
28.0855
>>> si.melting_point
u'1687 K'
```

If the installation worked correctly, the atomic mass and melting point of silicon should be displayed.

Updating

Due to the rapid development of pymatgen, you may someday want to upgrade to the newest version. Updating from a package can be done with the same command that was used for installing, while upgrading from the internet can be done with the following command:

```
python -measy_install -U pymatgen
```

> **❗ Important**
>
> The pymatgen change-log should be checked before upgrading, to ensure that all changes are compatible with current code, e.g. scripts and plugins.