

EGU21-13096

<https://doi.org/10.5194/egusphere-egu21-13096>

EGU General Assembly 2021

© Author(s) 2022. This work is distributed under the Creative Commons Attribution 4.0 License.



## pyam - an open-source Python package for energy system scenario analysis & visualization

Daniel Huppmann<sup>1</sup> and Matthew Gidden<sup>2</sup>

<sup>1</sup>International Institute for Applied Systems Analysis (IIASA)

<sup>2</sup>Climate Analytics

### Background: Transparency, reproducibility and reusability of scientific analysis

Researchers are now widely expected to share the data and source code of their work to foster transparency, reproducibility and reusability.

Alas, the quality of data documentation and scientific software scripts can vary substantially. In many instances, metadata and information on the provenance of data are missing or incomplete, and source code often does not include a clear list of dependencies (including version information) or systems requirements. Finally, the source code does not include sufficient inline documentation to be easily understood. As a consequence, even though the data and related scripts may be released under an open-source license, analysis too often cannot be reproduced or adapted with reasonable effort by other researchers.

### The pyam package

This talk presents the open-source Python package **pyam** for energy system scenario analysis and visualization. The aim of pyam is not to provide any ground-breaking new methods or analysis routines. Instead, it provides a reliable, well-tested interface similar in feel & style to the widely used **pandas** package, but geared for data formats and applications often used in energy systems analysis and integrated assessment modelling.

By using pyam for their scenario input data processing and analysis workflows, researchers can reduce standard tasks like unit conversion and data validation from a 5-minute effort to 30 seconds - and have the knowledge that their scripts won't break if pandas or another dependency change their APIs, because the pyam community will work to ensure forward-compatibility and continuity. As another side benefit, the pyam package will raise meaningful errors when input data doesn't make sense, whereas own ad-hoc scripts may fail silently or - even worse - return non-sensical values.

### Spatial, temporal and sectoral aggregation & downscaling features

To highlight the applicability of the pyam package for the EGU community and energy & climate modellers at large, this talk will focus on the features for spatial, temporal and sectoral

aggregation and downscaling. The package include several often-used methods like weighting by proxy variables or deriving indicators based on minimum or maximum values of timeseries data.

### **Building a community**

The pyam package follows best-practice of version control, continuous-integration and scientific-software documentation. This facilitates building on the package by other researchers. The community uses several tools for communication and discussion, including a Slack channel, an email list and a Github repository for issues & pull requests. And of course, we appreciate contributions by colleagues to extend the scope of features and methods based on their own use cases and requirements!

### **More information**

ReadTheDocs: <https://pyam-iamc.readthedocs.io/>

GitHub repo: <https://github.com/iamconsortium/pyam>