The Journal of Open Source Software

SpiceyPy: a Pythonic Wrapper for the SPICE Toolkit

Andrew M. Annex[1], **Ben Pearson**[2], **Benoît Seignovert**[3], **Brian T. Carcich**[4], **Helge Eichhorn**[5], **Jesse A. Mapel**[6], **Johan L. Freiherr von Forstner**[7], **Jonathan McAuliffe**[8], **Jorge Diaz del Rio**[9], **Kristin L. Berry**[6], **K.-Michael Aye**[10], **Marcel Stefko**[11], **Miguel de Val-Borro**[12], **Shankar Kulumani**[13], and **Shin-ya Murakami**[14]

1 Johns Hopkins University **2** None **3** Jet Propulsion Laboratory, California Institute of Technology **4** Latchmoor Services, LLC **5** Planetary Transportation Systems GmbH **6** USGS Astrogeology Science Center **7** Institute of Experimental and Applied Physics, University of Kiel **8** DLR Gesellschaft für Raumfahrtanwendungen (GfR) mbH **9** ODC Space **10** Laboratory for Atmospheric and Space Physics, University of Colorado **11** ETH Zurich **12** Planetary Science Institute **13** Collins Aerospace **14** GFD Dennou Club

DOI: 10.21105/joss.02050

Software

- Review
- Repository
- Archive

Editor: Monica Bobra

Reviewers:

- @hayesla
- @jessie-dotson

Submitted: 24 January 2020
Published: 21 February 2020

Statement of Need

Operating in space necessitates quantifying the positions, velocities, geometries, and other properties of spacecraft and planetary bodies through time. Scientists and engineers working with robotic planetary spacecraft missions use the Spacecraft, Planet, Instrument, Camera-matrix, Events (SPICE) Toolkit (Acton, Bachman, Semenov, & Wright, 2018) to help plan observations and to quantify the positions of planetary bodies and spacecraft through time. SPICE is developed at the Jet Propulsion Laboratory by NASA's Navigation and Ancillary Information Facility (NAIF). Scientists also use SPICE to analyze data returned by these missions and to plan hypothetical orbital trajectories for future missions (Acton et al., 2018). For example, SPICE can calculate future occultations of planets relative to a camera on a rover or spacecraft. The NAIF provides SPICE in Fortran 77, C, and they also provide Matlab and IDL wrappers; however, as of 2014, they did not offer a Python interface. The growth of Python and movement away from proprietary interpreted languages (Burrell et al., 2018) motivated the development of SpiceyPy so that planetary scientists and engineers can use SPICE within Python.

Summary

`SpiceyPy` is an open-source, MIT licensed Python package that provides a `pythonic` interface to nearly all of the C SPICE toolkit N66. `SpiceyPy` was developed in Python using the `ctypes` module of the CPython standard library to wrap the underlying C SPICE shared library. Developing `SpiceyPy` in Python enabled the SpiceyPy API to expose simplified and more `pythonic` interactions with the underlying C API for SPICE. `SpiceyPy` relies on the NumPy library for numeric arrays and tight integration with the SciPy stack.

`SpiceyPy` is extensively tested using a combination of unit and integration tests, which run using continuous integration services. The tests also serve as code examples translated from the NAIF documentation. Continuous deployment updates documentation and deploys artifacts of releases to PyPI and the conda-forge. Every SPICE function wrapper in SpiceyPy contains docstrings that provide short descriptions of the function duplicated from the SPICE documentation. Docstrings in SpiceyPy also contain links to the corresponding CSPICE documentation page hosted by the NAIF to provide additional details regarding the function.

SpiceyPy enables scientists to utilize the full functionality of SPICE within Python and the ecosystem of visualization and scientific packages available. SpiceyPy has been utilized in peer-reviewed research (Attree et al., 2019; Behar, Nilsson, Alho, Goetz, & Tsurutani, 2017; Behar et al., 2016; Porter et al., 2018; Zangari, Finley, Alan Stern, & Tapley, 2018), masters and doctoral theses (Albin, 2019; Hackett, 2019), spacecraft mission operations, as a dependency in other Python libraries (Stansby, Rai, Broll, Shaw, & others, 2019), and for a variety of other projects (Costa & Grass, 2018; Wilson, 2017; Wilson & Xiong, 2016).

Acknowledgements

The authors would like to acknowledge members of the NAIF (Charles Acton, Ed Wright, Boris Semenov, Nat Bachman) for continued support for SpiceyPy and for providing to users a *SpiceyPy translation* of their excellent "Hands-on" lessons. The first author also thanks all of the contributors and users of SpiceyPy; they motivate further improvements to the project. Co-authors other than the first author are ordered solely alphabetically by their first name.

References

Acton, C., Bachman, N., Semenov, B., & Wright, E. (2018). A look towards the future in the handling of space science mission geometry. *Planetary and Space Science*, *150*(2018), 9–12. doi:10.1016/j.pss.2017.02.013

Albin, T. (2019). *Machine learning and monte carlo based data analysis methods in cosmic dust research* (PhD thesis). University of Stuttgart.

Attree, N., Jorda, L., Groussin, O., Mottola, S., Thomas, N., Brouet, Y., Kührt, E., et al. (2019). Constraining models of activity on comet 67P/churyumov-gerasimenko with rosetta trajectory, rotation, and water production measurements. *Astronomy and astrophysics*. doi:10.1051/0004-6361/201834415

Behar, E., Nilsson, H., Alho, M., Goetz, C., & Tsurutani, B. (2017). The birth and growth of a solar wind cavity around a comet–rosetta observations. *Monthly Notices of the Royal Astronomical Society*, *469*(Suppl_2), S396–S403. doi:10.1093/mnras/stx1871

Behar, E., Nilsson, H., Wieser, G. S., Nemeth, Z., Broiles, T., & Richter, I. (2016). Mass loading at 67P/churyumov-gerasimenko: A case study. *Geophysical Research Letters*, *43*(4), 1411–1418. doi:10.1002/2015gl067436

Burrell, A. G., Halford, A., Klenzing, J., Stoneback, R., Morley, S. K., Annex, A., Laundal, K., et al. (2018). Snakes on a spaceship—an overview of python in heliophysics. *Journal of Geophysical Research: Space Physics*, *123*(12), 10–384. doi:10.1029/2018ja025877

Costa, M., & Grass, M. (2018). SPICE-based python packages for solar system exploration geometry exploitation. In *European planetary science congress* (Vol. 12, pp. EPSC2018–53). Retrieved from https://ui.adsabs.harvard.edu/abs/2018EPSC...12...53C

Hackett, T. M. (2019). *Applying artificial intelligence to space communications networks: Cognitive real-time link layer adaptations through rapid orbit planning* (PhD thesis). The Pennsylvania State University. Retrieved from https://etda.libraries.psu.edu/catalog/16634tmh5344

Porter, S. B., Buie, M. W., Parker, A. H., Spencer, J. R., Benecchi, S., Tanga, P., Verbiscer, A., et al. (2018). High-precision Orbit Fitting and Uncertainty Analysis of (486958) 2014 MU69. *The Astronomical Journal*, *156*(1), 20. doi:10.3847/1538-3881/aac2e1

Stansby, D., Rai, Y., Broll, J., Shaw, S., & others. (2019). HelioPy: Heliospheric and planetary physics library. *Astrophysics Source Code Library*. doi:10.5281/zenodo.1009079

Wilson, C. (2017). See how the solar eclipse will look from anywhere in the u.s. *Time Magazine*. Retrieved from https://time.com/4882923/total-solar-eclipse-map-places-view/

Wilson, T., & Xiong, X. (2016). Scheduling observations of celestial objects for Earth observing sensor calibration. In R. Meynart, S. P. Neeck, T. Kimura, & H. Shimoda (Eds.), *Sensors, Systems, and Next-Generation Satellites XX*. SPIE. doi:10.1117/12.2240648

Zangari, A. M., Finley, T. J., Alan Stern, S., & Tapley, M. B. (2018). Return to the kuiper belt: Launch opportunities from 2025 to 2040. *Journal of Spacecraft and Rockets*, *56*(3), 919–930. doi:10.2514/1.a34329