exoatlas: friendly Python code for exoplanet populations

Zach K. Berta-Thompson [1], **Patcharapol Wachiraphan** [1], **Autumn Stephens** [1], **Mirielle Caradonna** [1], **Catriona Murray** [1], **Valerie Arriero** [1], **Jackson Avery** [1], **Girish M. Duvvuri** [2], and **Sebastian Pineda** [3]

1 Department of Astrophysical and Planetary Sciences, University of Colorado Boulder, USA ROR **2** Department of Physics and Astronomy, Vanderbilt University, USA ROR **3** Laboratory for Atmospheric and Space Physics, University of Colorado Boulder, USA ROR

Summary

Planets are complicated. Understanding how they work requires connecting individual objects to the context of broader populations. Exoplanets are easier to picture next to their closest Solar System archetypes, and planets in the Solar System are richer when seen alongside a growing community of known exoplanets in the Milky Way. The exoatlas toolkit provides a friendly Python interface for retrieving and working with populations of planets, aiming to simplify the process of placing worlds in context.

Statement of need

exoatlas was designed to meet a need among both researchers and educators for an intuitive Python tool to access planet populations. Particularly in working with students and junior scientists, for whom easy avenues for exploration and play would have particular benefit, significant barriers often frustrate attempts to perform the following tasks:

- retrieving basic properties for exoplanets + Solar System objects
- calculating derived planet quantities with propagated uncertainties
- comparing individual exoplanets to relevant comparison samples
- planning future telescope observations of known exoplanet systems
- making beautiful and up-to-date planet data visualizations

Online planetary data archives merge incredible curatorial efforts with powerful tools for data access and visualization, including the NASA Exoplanet Archive (Christiansen et al., 2025), exoplanet.eu (exoplanet.eu, 2025), exo.MAST (STScI, 2025), TEPCat (Southworth, 2011), Open Exoplanet Catalog (Rein, 2012) for exoplanets and the JPL Solar System Dynamics (JPL, 2025), IAU Minor Planet Center (IAU, 2025), the NSSDCA Planetary Facts sheets (Williams, 2025) for Solar System objects. exoatlas does not intend to replace any of these important archival efforts (it pulls nearly all its data from them); rather, exoatlas aims to provide an approachable interface for exploratory analysis and illuminating visualizations to help the community make better use of these resources.

Mapping populations with `exoatlas`

The user interface for exoatlas centers around the Python Population class, with each Population object containing a standardized table of planet properties and methods for interacting with that table. exoatlas makes extensive use of astropy (Astropy Collaboration

et al., 2022) to be as familiar as possible for modern astronomers, and it is thoroughly documented with astronomical audiences in mind at zkbt.github.io/exoatlas/.

- To retrieve planet data, exoatlas provides `Population` objects that automatically access archive data for exoplanets as well as Solar System major planets, minor planets, and moons. Exoplanet data come from the NASA Exoplanet Archive (Christiansen et al., 2025) API, and Solar System data come from the JPL Solar System Dynamics (JPL, 2025) API or small reformatted data tables included in the package repository itself. Whatever the original data source, all `Population` objects act similarly and have uniform nomenclature for accessing data columns. Planet quantities all have physical units attached with `astropy.units` to facilitate unit conversions and minimize conceptual errors. New `Population` objects can also be created from `astropy.table` tables, enabling custom datasets to be included.

- To calculate derived quantities for planets, a set of default methods are included within the core `Population` definition, or users may attach their own calculation methods. If data for a quantity is missing, calculations can be used to swap in alternate estimates; for example, a planet's semimajor axis a will attempt first to pull from the original data table, and then second to calculate a from the planet's period P and the star's mass $M_\star$ assuming Newton's Version of Kepler's Third Law $P^2 = 4\pi^2 a^3/GM_\star$, and then third to calculate from a transit-derived ratio a/R_*. Uncertainties on all derived quantities can be numerically propagated using the `astropy.uncertainty` framework, where distributions of samples are generated for each original table quantity, carried through calculations, and then used to estimate confidence intervals.

- To extract sets of planets meeting particular criteria, `Population` objects can be indexed, sliced, and masked to generate new smaller `Population` objects. Coupling familiar array or table operations into the creation of subpopulations enables nuanced filtering of datasets based on any combination of original archival table quantities, derived quantities, and/or quantity uncertainties.

- To plan telescope observations of known exoplanet systems, exoatlas can estimate the signal-to-noise (S/N) ratio acheivable for exoplanet observables. As derived quantities, all S/N estimates can include propagated uncertainties, to help avoid biasing target samples toward systems with larger uncertainties. Using `astroplan` (Morris et al., 2018) and `astropy.coordinates` (Astropy Collaboration et al., 2022), exoatlas can determine local altitudes and azimuths for all elements in a `Population`, as well as upcoming opportunities to observe transits from ground-based telescopes (when the star is above the horizon, the Sun is down, and the planet is passing in front of the star). Including basic observation planning tools enables a workflow where targets can be filtered both by the estimated detectability of a signal and by whether a telescope can actually point at the system.

- To make explanatory illustrations of planet data, exoatlas includes a visual language to build up population comparison plots. The core elements of this visual language are the `Plottable` (a quantity that should be represented with certain scaling, limits, and labels), the `Map` (a panel expressing plottable quantities with position, size, or color), and the `Gallery` (a collection of maps with linked axes and/or datasets). Preset visualizations built from this language can offer quick contextual reference for a planet's fundamental properties, as in Figure Figure 1, which was generated with only six lines of code:

```python
from exoatlas import *
from exoatlas.visualizations import *

e = TransitingExoplanets()
s = SolarSystem()
h = e["HD209458b"]

PlanetGallery().build([e, s, h])
```
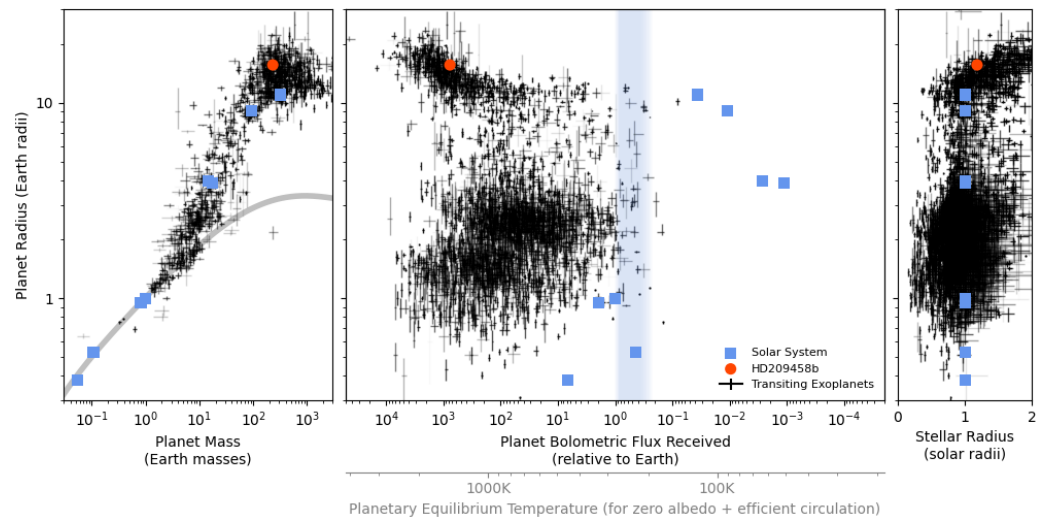
Figure 1: Example exoatlas visualization placing the first discovered transiting exoplanet HD209458b in context with other transiting exoplanets and the eight major Solar System planets. Errorbars use a color intensity that scales inversely with quantity uncertainties, to avoid giving undue visual weight to the least precise data.

Research, teaching, and learning

exoatlas was designed to support the researcher who wants to contextualize planet populations in papers, proposals, or talks, the educator who wants help connect lessons more immediately to real exoplanet and Solar System data, and the student who simply wants to learn a little more about oodles of neat planets. All these communities are encouraged try exoatlas, to ask for help, to suggest improvements, and/or to contribute code!

Acknowledgements

References

Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L., Shupe, D. L., Patil, A. A., Corrales, L., Brasseur, C. E., Nöthe, M., Donath, A., Tollerud, E., Morris, B. M., Ginsburg, A., Vaher, E., Weaver, B. A., Tocknell, J., Jamieson, W., … Astropy Project Contributors. (2022). The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package. *935*(2), 167. https://doi.org/10.3847/1538-4357/ac7c74

Christiansen, J. L., McElroy, D. L., Harbut, M., Ciardi, D. R., Crane, M., Good, J., Hardegree-Ullman, K. K., Kesseli, A. Y., Lund, M. B., Lynn, M., Muthiar, A., Nilsson, R., Oluyide, T., Papin, M., Rivera, A., Swain, M., Susemiehl, N. D., Tam, R., van Eyken, J., & Beichman, C. (2025). The NASA Exoplanet Archive and Exoplanet Follow-up Observing Program: Data, Tools, and Usage. *arXiv e-Prints*, arXiv:2506.03299. https://doi.org/10.48550/arXiv.2506.03299

exoplanet.eu. (2025). *Exoplanet.eu*. https://exoplanet.eu/catalog/

109 IAU. (2025). *Minor planet center*. https://minorplanetcenter.net

110 JPL. (2025). *Solar system dynamics*. https://ssd.jpl.nasa.gov

111 Morris, B. M., Tollerud, E., Sipőcz, B., Deil, C., Douglas, S. T., Berlanga Medina, J.,
112 Vyhmeister, K., Smith, T. R., Littlefair, S., Price-Whelan, A. M., Gee, W. T., & Jeschke,
113 E. (2018). astroplan: An Open Source Observation Planning Package in Python. *155*(3),
114 128. https://doi.org/10.3847/1538-3881/aaa47e

115 Rein, H. (2012). A proposal for community driven and decentralized astronomical databases
116 and the Open Exoplanet Catalogue. *arXiv e-Prints*, arXiv:1211.7121. https://doi.org/10.
117 48550/arXiv.1211.7121

118 Southworth, J. (2011). Homogeneous studies of transiting extrasolar planets - IV. Thirty
119 systems with space-based light curves. *417*(3), 2166–2196. https://doi.org/10.1111/j.
120 1365-2966.2011.19399.x

121 STScI. (2025). *Exo.MAST*. https://exo.mast.stsci.edu

122 Williams, D. R. (2025). *NSSDCA planetary fact sheets*. https://nssdc.gsfc.nasa.gov/planetary/
123 planetfact.html