The Journal of Open Source Software

SkyPortal: An Astronomical Data Platform

Stéfan J. van der Walt[1], **Arien Crellin-Quick**[2], **and Joshua S. Bloom**[2]

1 Berkeley Institute for Data Science, University of California, Berkeley **2** Department of Astronomy, University of California, Berkeley

DOI: 10.21105/joss.01247

Software

- Review
- Repository
- Archive

Submitted: 09 February 2019
Published: 12 May 2019

Summary

SkyPortal is a web application that stores and interactively displays astronomical datasets for annotation, analysis, and discovery. It is designed to be modular and extensible, so it can be customized for various scientific use cases. It is released under the Modified BSD license.

SkyPortal was designed with time-series survey data from the Zwicky Transient Facility, and eventually The Large Synoptic Survey Telescope, in mind.

By default, it aims to provide a useful, rich user experience, including light curves of named astronomical events/transients, spectra, live chat, and links to other surveys. But the intent, ultimately, is for the frontend to be modified to best suit the specific scientific problem at hand. The current UI/UX was inspired by that developed for the Palomar Transient Factory (PTF) Marshal (Law et al., 2009).

Architecture

SkyPortal builds on top of baselayer, a customizable scientific web application platform developed by the same authors as part of the Cesium-ML time-series project (Naul, Van der Walt, Crellin-Quick, Bloom, & Pérez, 2016). Baselayer provides SkyPortal with authentication (via OAuth, Google, etc.), websockets (to communicate between the Python backend and the JavaScript frontend), and the scaffolding for managing microservices and routing incoming requests via Nginx.

The application has a Python backend (running the Tornado web server), with a React & Redux frontend. React was chosen because of its clean component design, and Redux provides the application logic that renders these components appropriately, given the application state.

For machine users, SkyPortal provides a token-based API, meaning that all of its data can be queried and modified by scripts without using the browser frontend.

Importantly, the application is able to provide graphical renditions of datasets, using the Bokeh library. In the default version of SkyPortal, this functionality is used to render, e.g., spectra, with the ability to toggle color bands and element spectra, or to adjust redshift.

The platform also has the ability to perform distributed computation via Dask.

SkyPortal implements two types of security: group based, and Access Control List (ACL) based. Group based security determines which users have access to which sources (data objects), each of which is associated with one or more group. The members of these

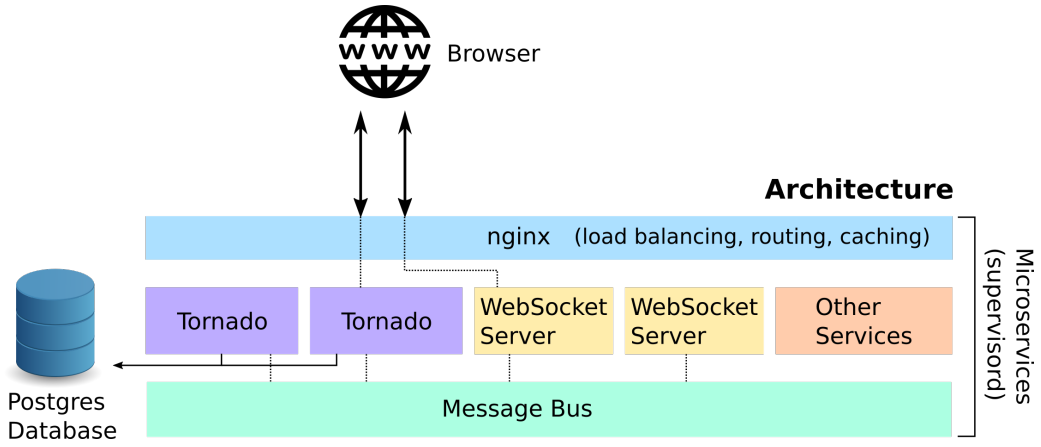

Figure 1: SkyPortal architecture. Each technological component is chosen to be efficient and scalable, and is applied exactly for the purpose it was built.

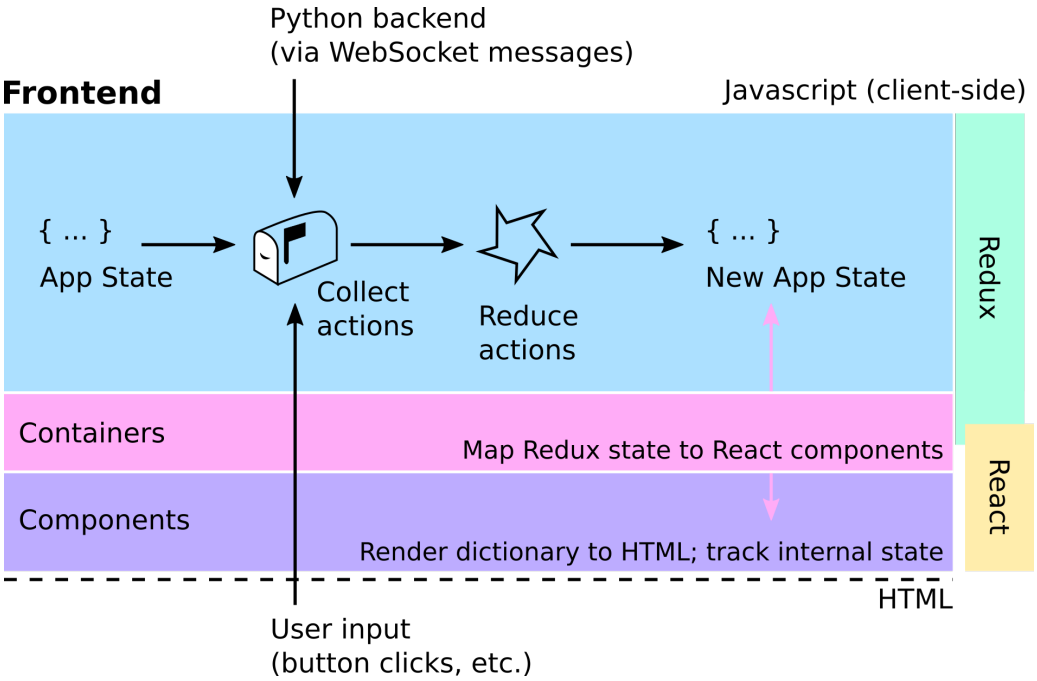

Figure 2: Frontend architecture. Components are rendered by React, and application logic is driven by Redux.

groups can be changed by the administrator. ACL based security deals with user roles, which determine, regardless of data source, which pages a user can access.

SkyPortal is designed to be employed on systems of varying scale: a laptop, local server, or hosted infrastructure. We support packaging the entire application into a Docker container, which can then be deployed to the cloud. An example deployment, using Kubernetes, is provided.

Integration testing is done upon every commit, using Travis-CI, by automating the Firefox browser with GeckoDriver. This ensures that the entire user experience—from logging in to making API requests—keeps working correctly.

Future work

In the next version of SkyPortal, we intend to support loading sources in real-time from astronomical surveys, typically by ingesting Kafka streams. We also plan to add the ability to publish processed sub-streams, creating so-called "brokers" (Swinbank, 2014). Finally, it will be possible to customize views (from the web interface), and for developers to easily add and modify user-interface components.

Conclusion

SkyPortal is an extensible data platform for astronomy. It is currently being used to analyze data from various sky surveys, with the potential to aid many more astronomers in their data processing and visualization needs. It is actively developed and maintained, and the authors welcome and encourage collaboration.

Acknowledgements

We thank M. Kasliwal and D. Goldstein for helpful discussions and insights during the prototyping and build-out phases of this project. This work was supported by a Gordon and Betty Moore Foundation Data-Driven Discovery grant.

References

Law, N. M., Kulkarni, S. R., Dekany, R. G., Ofek, E. O., Quimby, R. M., Nugent, P. E., Surace, J., et al. (2009). The Palomar Transient Factory: System Overview, Performance, and First Results. *Publications of the Astronomical Society of the Pacific*, *121*, 1395. doi:10.1086/648598

Naul, B., Van der Walt, S. J., Crellin-Quick, A., Bloom, J. S., & Pérez, F. (2016). Cesium: Open-Source Platform for Time-Series Inference. In Sebastian Benthall & Scott Rostrup (Eds.), *Proceedings of the 15th Python in Science Conference* (pp. 27–35). doi:10.25080/Majora-629e541a-004

Swinbank, J. (2014). Comet: A VOEvent broker. *Astronomy and Computing*, *7*, 12–26. doi:10.1016/j.ascom.2014.09.001

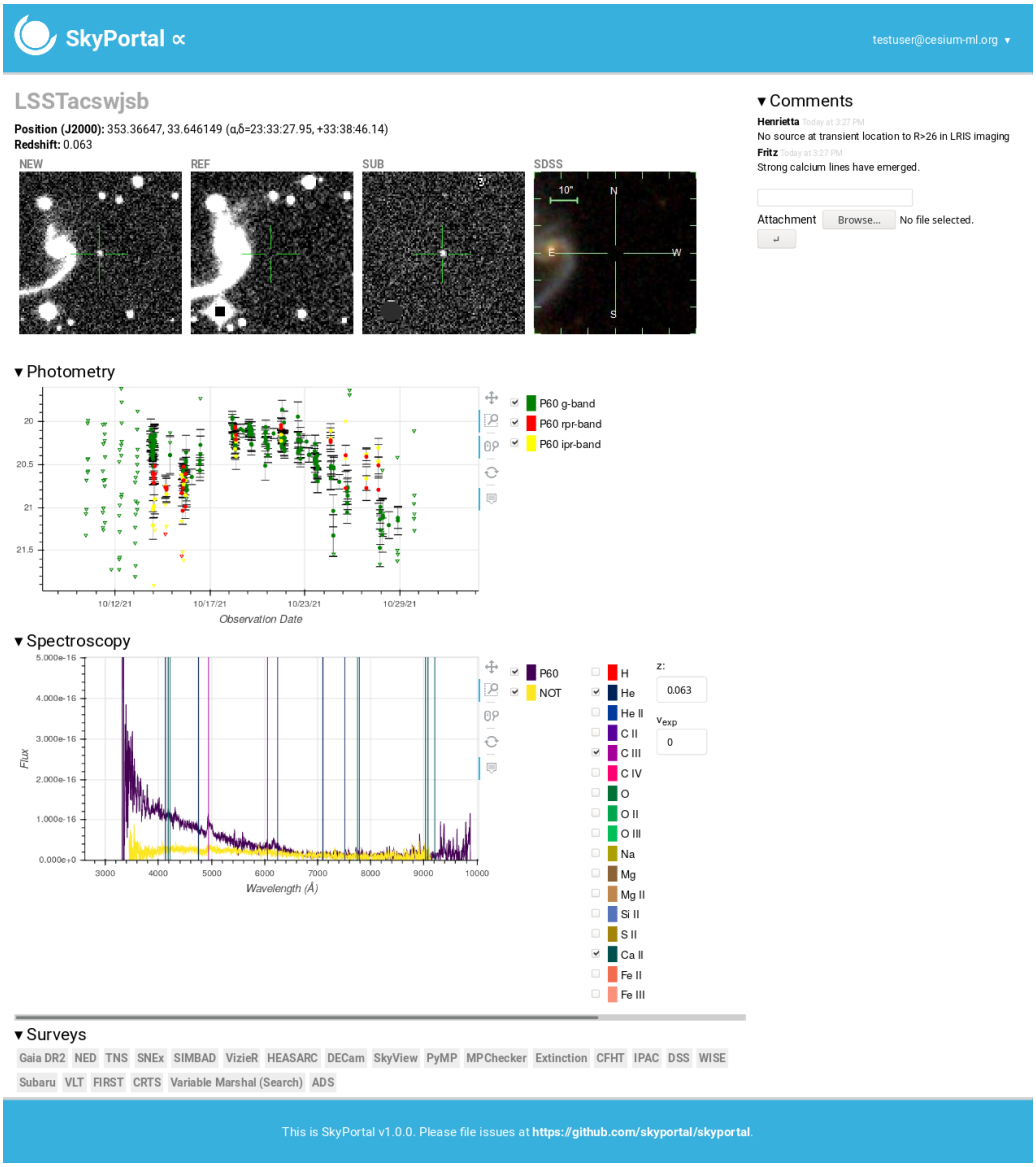

Figure 3: SkyPortal source rendering. Featured are thumbnails, light curve plot, spectrogram, live chat, and links to various sky surveys.