SODAR Core: a Django-based framework for scientific data management and analysis web apps

Mikko Nieminen[1], **Oliver Stolpe**[1], **Franziska Schumann**[1], **Manuel Holtgrewe**[1], **and Dieter Beule**[1]

1 Berlin Institute of Health, Anna-Louisa-Karsch-Straße 2, 10178 Berlin

DOI: 10.21105/joss.01584

Software

- Review
- Repository
- Archive

Editor: Lorena Pantano

Reviewers:

- @rbatorsky
- @olgabot

Submitted: 02 July 2019
Published: 03 February 2020

Introduction

Modern life science is generating large data sets at a unprecedented speed. A major source of data are so-called omics (e.g., genomics, metabolomics, or proteomics) experiments. Consequently, management and analysis of scientific data has become a major challenge. Further, the heterogeneity of projects makes "one size fits all" data analysis systems infeasible and calls for specialized data analysis platforms.

The authors are actively developing applications for the FAIR (findable, accessible, interoperable, and reuseable, cf. Wilkinson et al. (2016) data management of omics data and their analysis. In order to prevent duplication of work, we have extracted the commonly useful components into *SODAR Core*, a Python framework on top of Django. It is used in our actively developed applications and also proved useful in internal web app prototypes.

Examples for using SODAR Core in the development of scientific data management web apps is Digestiflow (Holtgrewe, Nieminen, Messerschmidt, & Beule, 2019) and the Filesfolders module shipping with SODAR Core. An example for using SODAR Core in the development of scientific data analysis web apps is VarFish (Holtgrewe et al., 2020).

Features

SODAR Core provides its features in the form of Django apps. This section lists the main applications at the time of writing and summarises their features.

Projects and project-based access control

All SODAR Core features are grouped around projects that can be arranged in a tree structure. Users can be assigned to projects with certain roles that can be used for controlling authorization. This feature is provided in the `projectroles` app. Further, SODAR Core allows for easy integration of more than one external LDAP authentication source.

SODAR Core currently comes with a fixed set of roles that we found a appropriate for a number of projects. Making the role set configurable is a part of our future development roadmap.

Common Projects in Separated Applications

As mentioned above, we found that the complexity of omics datasets requires specialized data analysis applications. There are several advantages of separating and decoupling these specialized apps from the main data management system. These include the ability for having different rates of development and stability, e.g., having a separate core data management system and experimental and less well-tested data analysis platforms. We also found the ability to reduce dependency of development teams from each other very useful.

On the other hand, project and authorization generally should be shared between the data management and data analysis projects. To prevent users from having to manually replicate projects and authorization configuration SODAR Core provides a *remote site* feature. An application instance can either be a *source* or a *target* site. Target sites can be connected to a source site and obtain project and authorization information from the source site.

Auditing and Timelines

The `timeline` app provides an API for storing events in a project-specific timeline. This allows for implementing audit trails and a history or changelog. We have found this very useful to track changes done by users and keep a provenance trail.

Asynchronous Background Jobs

In many data analysis use cases, there will be a number of data processing tasks that take a longer time, e.g., a complex database query, copying large volumes of data, or longer numeric computation. In the context of web applications, it is important to implement them as asynchronous jobs as implemented in the `bgjobs` app. This allows the web app to render HTML documents to the client's browser independently of completing the data analysis task.

Caching Infrastructure

In many cases, data management and analysis applications will use data stored in external services that might take a longer time to query. For this use case, the `sodarcache` app provides features for caching query results.

Miscellaneous Functionality

Further, there are a number of apps in SODAR core that help with various recurrent tasks. The `userprofile` provide user profiles and together with `projectroles` it allows storing user and/or project-specific settings. `siteinfo` allows to create a site-wide statistics view while `adminalerts` allows to display and schedule site-wide notifications, e.g., for announcing maintenances.

In addition, we have found the management of files (in the range of several MB) useful for many applications such as sharing documents or storing attachments to messages. The app `filesfolders` provides the feature that allows users to upload files and manage them in a folder tree structure.

Summary and Conclusion

SODAR Core addresses many recurring requirements for scientific data management and analysis projects. The authors are actively using it in their projects. We believe that other people will find the components useful in their projects and are thus publishing them as free open source software.

License and Availability

SODAR Core is distributed under the MIT license and available from Github at https://github.com/bihealth/sodar_core. Each release is also stored to Zenodo. The current version 0.6.2 is available with the DOI 10.5281/zenodo.3251782. An example Django site is contained in the SODAR Core Git repository and up-to date documentation is available at https://sodar-core.readthedocs.org.

References

Holtgrewe, M., Nieminen, M., Messerschmidt, C., & Beule, D. (2019). DigestiFlow - reproducible demultiplexing for the single cell era. *PeerJ Preprints*, *7*, e27717v2. doi:10.7287/peerj.preprints.27717v2

Holtgrewe, M., Stolpe, O., Nieminen, M., Mundlos, S., Knaus, A., Kornak, U., Seelow, D., et al. (2020). VarFish - collaborative and comprehensive variant analysis for diagnosis and research. *bioRxiv*. doi:10.1101/2020.01.27.921965

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., et al. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data*, *3*, 160018. doi:10.1038/sdata.2016.18