OPEN ASTRONOMY CATALOGS API

James Guillochon1 and Philip S. Cowperthwaite1

1 Harvard-Smithsonian Center for Astrophysics
60 Garden St.
Cambridge, MA 02138, USA

Keywords: editorials, notices — catalogs

DESCRIPTION

In this research note we announce the public release of the application program interface (API) for the Open Astronomy Catalogs (OACs, Guillochon et al. 2017), the OACAPI. The OACs serve near-complete collections of supernova2, tidal disruption3 (Auchettl et al. 2017), kilonova4 (Villar et al. 2017), and fast stars5 (Boubert et al. 2018) data (including photometry, spectra, radio, and X-ray observations) via a user-friendly web interface that displays the data interactively and offers full data downloads. The API, by contrast, enables users to specifically download particular pieces of the OAC dataset via a flexible programmatic syntax.

While the description presented in this note is current with the active version of the API (1.0), a living version of this description is available on the git repository for the API software6, the reader is encouraged to bookmark that document to keep abreast of any future API changes. Note that no API key is presently required for access. Users are also able to run the API locally on their own computers and/or servers, in case a dedicated copy of the API is desired7.

REST API

The API is implemented via a Representational State Transfer (REST) service that allows users to access OAC data via GET requests (POST support coming soon), where the primary route to the data is in the form

https://api.DOMAIN/OBJECT1+OBJECT2+.../QUANTITY1+QUANTITY2+.../
ATTRIBUTE1+ATTRIBUTE2+...?ARGUMENT1=VALUE1&ARGUMENT2=VALUE2&...

where DOMAIN corresponds to the catalog corresponding to the object type of the user’s interest (e.g. sne.space for supernovae, tde.space for tidal disruptions, etc.), OBJECT is set to a + delimited list of object names, QUANTITY is set to a + delimited list of quantities to retrieve from all objects, ATTRIBUTE is a + delimited list of properties of those quantities, and the ARGUMENT variables (delimited by &) allow to user to filter data based upon various attribute values.

Alternatively, a user can locate objects by replacing the list of objects with catalog

https://api.DOMAIN/catalog/QUANTITY1+QUANTITY2+.../
ATTRIBUTE1+ATTRIBUTE2+...?ARGUMENT1=VALUE1&ARGUMENT2=VALUE2&...

which will perform a search across the full catalog to retrieve the requested data. Routes that exclude the list of objects and/or quantities and/or attributes are also valid; the API will return all data corresponding to the full list of the omitted route components in these cases (see the examples in Table 1).

Key names that are usable in API calls (for the QUANTITY and ATTRIBUTE parts of the API route) can be found in the OAC schema8. The ARGUMENT variables can be used to guarantee that a certain attribute appears in the

---

1 Source code available at https://github.com/astrocatalogs/OACAPI
2 https://sne.space
3 https://tde.space
4 https://kilonova.space
5 https://faststars.space
6 https://github.com/astrocatalogs/OACAPI
7 A service setup for Linux is available at https://gist.github.com/guillochon/7148fe7a310cd4d08657f7e61c98dfe9
8 https://github.com/astrocatalogs/schema
returned results (e.g. adding \texttt{\&time\&e.magnitudes} to the query will guarantee that each returned item has a time and e.magnitudes attribute, and/or used to filter via a simple equality such as \texttt{telescope=HST} (which would only return QUANTITY objects where the telescope attribute equals “HST”), and/or matched against regular expressions, and/or used for more sophisticated operations (an example being cone searches \texttt{ra} and \texttt{dec}). Below we show a list of special attributes that can be used to perform advanced queries:

- \texttt{closest}: Return the quantities with the closest value to the specified attributes. If multiple attributes are specified, the closest to each will be returned (e.g., \texttt{magnitude=15\&time=56789\&closest} would return both the observation with magnitude closest to 15 and time closest to 56789.
- \texttt{complete}: Return only quantities containing all of the requested attributes.
- \texttt{first}: Return only the first of each of the listed quantities.
- \texttt{format=x}: Return data in the specified format \texttt{x}, currently supports CSV and TSV. Any other format specification will return JSON.
- \texttt{item=n}: Return only the \texttt{n}th item of each of the listed quantities.
- \texttt{radius=r}: Return objects within a distance \texttt{r} (in arcseconds) of a given set of \texttt{ra} and \texttt{dec} coordinates. Note that this disables exact matches for \texttt{ra} and \texttt{dec}.
- \texttt{width=w}: Return objects within a distance \texttt{w} (in arcseconds) of a given \texttt{ra} value (for box searches).
- \texttt{height=h}: Return objects within a distance \texttt{h} (in arcseconds) of a given \texttt{dec} value (for box searches).
- \texttt{sortby=s}: Sort the returned array by the attribute \texttt{s} (only works when returning results in CSV or TSV formats).

In Table 1, we provide some example queries that demonstrate the API’s capabilities, these examples are hopefully useful to the reader as a starting point for OACAPI interactions.

### Astroquery Module

Along with the release of the REST API, we also announce the release of a module for the Astroquery package (part of the Astropy package, Astropy Collaboration et al. 2013), available in the 0.3.8 release of Astroquery\textsuperscript{9}. This module is designed to bring the full functionality of the REST API described above into a Pythonic workflow. The primary methods available to the user are:

- \texttt{query\_object}: This method returns the requested QUANTITIES and ATTRIBUTES for a specified OBJECT or list of OBJECTS.
- \texttt{query\_region}: This method returns the results of a cone or box search for a single set of coordinates. Results can be filtered based on desired QUANTITIES and ATTRIBUTES.

Users can also quickly obtain results using the following tailored methods:

- \texttt{get\_photometry}: This method returns all available photometry for a single OBJECT or list of OBJECTS.
- \texttt{get\_single\_spectrum}: This method returns a single spectrum for a single object at a specified MJD.
- \texttt{get\_spectra}: This method returns all available spectra for a single OBJECT or list of OBJECTS.

All of the methods described here return an Astropy table constructed from the returned CSV query. Users can also request a JSON compliant dictionary. We note that some searches, such as those returning multiple spectra, can not be processed into an Astropy table.

We thank the regular users of the Open Astronomy Catalogs for testing the API as it was being deployed, with special thanks to Sebastian Gomez for helpful comments. This work utilized the Astropy (Astropy Collaboration et al. 2013) and Astroquery (Ginsburg et al. 2018) packages.

\textsuperscript{9}https://github.com/astropy/astroquery/releases/tag/v0.3.8
| User wants                                                                 | URL route                                                                                     |
|---------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| All objects within a 2" cone about a set of coordinates                   | https://api.astrocats.space/catalog?ra=21:23:32.16&dec=-53:01:36.08&radius=2                  |
| Return all supernova metadata in CSV format                               | https://api.sne.space/catalog?format=csv                                                       |
| Redshifts of all supernovae with a redshift reported within 5° of a       | https://api.sne.space/catalog/redshift?ra=10:42:16.88&dec=-24:13:12.13&radius=18000&format=csv&redshift |
| coordinate, in CSV format                                                 |                                                                                               |
| Right ascensions and declinations of all objects with a redshift          | https://api.astrocats.space/catalog/ra+dec?ra=10:42:16.88&dec=-24:13:12.13&radius=18000&format=csv&redshift |
| reported within 5° of a coordinate, in CSV format                        |                                                                                               |
| Get all references listed for an object in CSV format                     | https://api.sne.space/SN2014J/sources/reference+bibcode?format=csv                           |
| Select the first (preferred) value of the redshift                       | https://api.astrocats.space/SN2014J/redshift?first or https://api.astrocats.space/SN2014J/redshift?item=0 |
| Return all photometric observations with at least one of the magnitude,  | https://api.astrocats.space/SN2014J/photometry/magnitude+emagnitude+band                      |
| e_magnitude, and band attributes                                          |                                                                                               |
| Return the above in CSV format                                           | https://api.astrocats.space/SN2014J/photometry/magnitude+emagnitude+band?format=csv           |
| Only return observations that contain all requested attributes            |                                                                                               |
| Sort the returned photometry by magnitude in CSV format                  | https://api.astrocats.space/SN2014J/photometry/magnitude+emagnitude+band?sortby=magnitude     |
| Return observations for multiple objects at once, in CSV format           | https://api.astrocats.space/SN2014J+SN2015F/photometry/time+band?format=csv                   |
| Return only observations whose attributes include the listed keys         | https://api.astrocats.space/SN2014J/photometry/time+magnitude+e_magnitude+band?complete       |
| (e_magnitude and band)                                                   |                                                                                               |
| Return only observations matching given criteria, in this case band = B   | https://api.astrocats.space/SN2014J/photometry/magnitude+e_magnitude+band?band=B              |
| Luminosity distances and claimed types of all objects with a available   | https://api.astrocats.space/catalog/lumdist+claimedtype?lumdist&claimedtype=ia&format=tsv      |
| luminosity distance and “Ia” listed as a type, in TSV format             |                                                                                               |
| Claimed types of all objects with a typing that has a prefix matching    | https://api.astrocats.space/catalog/claimedtype?claimedtype=Ia-(.*)&format=tsv                |
| “Ia-”, in TSV format                                                      |                                                                                               |
| Return the spectrum closest to the listed MJD                            | https://api.astrocats.space/SN2014J/spectra/time+data?time=56703.2&closest                   |
| Return all photometry in a 2" radius about a coordinate, in CSV format   | https://api.astrocats.space/catalog/photometry/time+band+magnitude?ra=21:23:32.16&dec=-53:01:36.08&radius=2&format=csv |
| Return the instruments used to produce spectra within a 5 of a given      | https://api.astrocats.space/catalog/spectra/instrument?ra=21:23:32.16&dec=-53:01:36.08&radius=18000&format=csv |
| coordinate, in CSV format                                                |                                                                                               |

Table 1. Example API queries to perform various data retrieval tasks from the OACs. A full list of retrievable quantities is available in the OAC schema.

REFERENCES

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33
Auchettl, K., Guillochon, J., & Ramirez-Ruiz, E. 2017, ApJ, 838, 149
Boubert, D., Guillochon, J., Hawkins, K., Ginsburg, I., & Evans, N. W. 2018, ArXiv e-prints, arXiv:1804.10179
Ginsburg, A., Sipocz, B., Parikh, M., et al. 2018, astropy/astroquery: v0.3.7 release, , , doi:10.5281/zenodo.1160027
Guillochon, J., Parrent, J., Kelley, L. Z., & Margutti, R. 2017, ApJ, 835, 64
Villar, V. A., Guillochon, J., Berger, E., et al. 2017, ApJL, 851, L21