



The SNAD Viewer: Everything You Want to Know about Your Favorite ZTF Object

Konstantin Malanchev^{1,2} , Matwey V. Kornilov^{2,3} , Maria V. Pruzhinskaya^{2,4} , Emille E. O. Ishida⁴ , Patrick D. Aleo^{1,5} , Vladimir S. Korolev⁶ , Anastasia Lavrukina⁷ , Etienne Russeil⁴ , Sreevarsha Sreejith⁸ , Alina A. Volnova⁹ , Anastasiya Voloshina⁴ , and Alberto Krone-Martins^{10,11} 

¹ Department of Astronomy, University of Illinois at Urbana-Champaign, 1002 W Green Street, Urbana, IL 61801, USA

² Sternberg Astronomical Institute, Lomonosov Moscow State University, Universitetsky pr. 13, Moscow 119234, Russia

³ National Research University Higher School of Economics, 21/4 Staraya Basmannaya Ulitsa, Moscow, 105066, Russia

⁴ Université Clermont Auvergne, CNRS/IN2P3, LPC, F-63000 Clermont-Ferrand, France

⁵ Center for AstroPhysical Surveys, National Center for Supercomputing Applications, Urbana, IL 61801, USA

⁶ Independent Researcher

⁷ Faculty of Space Research, Lomonosov Moscow State University, Leninsky Gori 1 bld. 52, Moscow 119234, Russia

⁸ Physics Department, Brookhaven National Laboratory, Upton, NY 11973, USA

⁹ Space Research Institute of the Russian Academy of Sciences (IKI), 84/32 Profsoyuznaya Street, Moscow, 117997, Russia

¹⁰ Donald Bren School of Information and Computer Sciences, University of California, Irvine, CA 92697, USA

¹¹ CENTRA/SIM, Faculdade de Ciências, Universidade de Lisboa, Ed. C8, Campo Grande, 1749-016, Lisboa, Portugal

Received 2022 November 23; accepted 2023 January 12; published 2023 February 27

Abstract

We describe the *SNAD Viewer*, a web portal for astronomers which presents a centralized view of individual objects from the Zwicky Transient Facility’s (ZTF) data releases, including data gathered from multiple publicly available astronomical archives and data sources. Initially built to enable efficient expert feedback in the context of adaptive machine learning applications, it has evolved into a full-fledged community asset that centralizes public information and provides a multi-dimensional view of ZTF sources. For users, we provide detailed descriptions of the data sources and choices underlying the information displayed in the portal. For developers, we describe our architectural choices and their consequences such that our experience can help others engaged in similar endeavors or in adapting our publicly released code to their requirements. The infrastructure we describe here is scalable and flexible and can be personalized and used by other surveys and for other science goals. The *Viewer* has been instrumental in highlighting the crucial roles domain experts retain in the era of big data in astronomy. Given the arrival of the upcoming generation of large-scale surveys, we believe similar systems will be paramount in enabling an optimal exploitation of the scientific potential enclosed in current terabyte and future petabyte-scale data sets. The *Viewer* is publicly available online at <https://ztf.snad.space>.

Unified Astronomy Thesaurus concepts: [Variable stars \(1761\)](#); [Astronomy software \(1855\)](#); [Open source software \(1866\)](#); [Publicly available software \(1864\)](#); [Astronomy web services \(1856\)](#)

1. Introduction

Modern astronomical surveys have drastically changed the process of astronomical data analysis. Traditionally, the field of astronomy was based on small data sets. Each of them gathered, analyzed, and reported by small research groups. Recently, the situation has drastically transformed into big data volumes, involving a large number of individuals at each stage of the process: from data ingestion to the final publication of scientific results. This long-predicted new paradigm (e.g., Szalay & Gray 2001) fostered the

development of crafted infrastructure within each survey, including strategies for storage, indexing, and archiving protocols specifically designed to fulfill the needs of a given scientific goal or community. The resulting data environments have enabled the successful application of machine learning techniques in astronomical data, especially for classification and regression tasks—two of the most well-known examples being photometric classification of supernovae (e.g., Jones et al. 2018; Vincenzi et al. 2022) and photometric redshift estimation (e.g., Zhou et al. 2021; Abbott et al. 2022). For such supervised learning tasks, once the experiment design and analysis pipeline is finalized, researchers can rerun the entire machinery within the same survey, updating results as more data become available, in principle, without any need for visual screening of the original data.



Original content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](#). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

At the same time, the observational nature of astronomy, coupled with the predetermined scanning strategies of large surveys, results in a tremendous potential for discovery (see, e.g., LSST-related reviews LSST Science Collaboration et al. 2009; Hambleton et al. 2022). In this context, human intervention is unavoidable. So far, even the most efficient machine learning algorithm cannot take into account all available data about each object it considers and can only provide good candidates as its output. These candidates usually need to be scrutinized by an expert who will put each of them in context and give meaning to a new discovery (Dick 2013). The expert, in turn, requires as much information as possible to form a comprehensive view of the new candidate and ensure its significance, as well as its novelty, reaching a final human conclusion. This means, for example, the researcher must gather data about the same object from multiple sources, build correlations between various wavelengths, and search for similar matches among well-known objects. In the specific case of transients or variables, one will also be interested in all available legacy data that can provide clues about their time evolution.

From the infrastructural point of view, this imposes a new set of requirements, which includes enabling easy visualization and cross-match between databases from different surveys, as well as with modern static catalogs, historical data, and derived data products. The SNAD team¹² met this challenge when our experts aimed to analyze a large set of variable objects. That is the reason why The SNAD Zwicky Transient Facility (ZTF) viewer was originally built (hereafter, the `Viewer`): the `Viewer` was designed and built to optimize the allocation of human resources in astronomical investigative tasks. It helps experts make decisions about the class and properties of a given object faster, consolidating photometry, cross-matching, and other information about astronomical sources retrieved from multiple data sets and data archives, on a single web page. Moreover, the `Viewer` is publicly openly available online at <https://ztf.snad.space>.

The time spent by an expert trying to mine information about an object became crucial when we started using active anomaly detection algorithms (Ishida et al. 2021). These human-in-the-loop learning strategies use feedback from a domain expert to guide adaptations of hyperparameters of a traditional machine learning model, thus constructing a model tailored to the expert’s own anomaly definition. The issue of scarce human resources for manual screening was already a limiting factor in the first stages of the SNAD pipeline development when we were dealing with a few thousand objects from the Open Supernova Catalog (Pruzhinskaya et al. 2019). However, it became unmanageable when faced with the Zwicky Transient Facility data releases¹³ (ZTF DR).

The ZTF is a time-domain photometric survey currently in progress using the Palomar 48 inch Schmidt telescope (Bellm et al. 2019a, 2019b). It represents the state of the art of large-scale astronomical surveys and is considered a precursor to the upcoming Vera C. Rubin Observatory Legacy Survey of Space and Time¹⁴ (LSST). For instance, data release 13, contains 4.37 billion light curves constructed from hundreds of billions of single-exposure extractions (see footnote 13).

There are a few publicly available tools that enable different levels of interaction with ZTF data, chief among them being the portals from ZTF community brokers (Section 2), IRSA IPAC,¹⁵ and Fritz Astronomy Marshal.¹⁶ Although they share some common features with the `Viewer`, broker systems were designed to deal with the data from the alert stream and to enable live analysis, while our goal was to explore the ZTF DRs. IRSA IPAC, on the other hand, provides a web user interface to access photometric data of surveys conducted by many different missions. Despite providing a convenient framework to handle ZTF DRs, its scope was not broad enough to fulfill the requirements of the SNAD anomaly detection pipeline. For expert analysis we required a portal providing instant access not only to a single object’s light curve, but also to as much relevant contextual information as possible, including nearby ZTF DR objects and cross-matched results from different catalogs. Moreover, IRSA IPAC gives access to the most recent data releases, while for the SNAD team, it was crucial to support a set of legacy data releases, whose data might be involved in ongoing projects. Fritz Astronomy Marshal is a ZTF platform produced by the project’s team and combining data from data releases, alert stream and cross-matching information from many catalogs. Access to the portal is restricted, but its source code is open and easy to run. However there is no easy way to ingest data-release photometry into the self-hosted instance of the system. Among non-ZTF frameworks with similar goals and infrastructure, we highlight the Open Supernova Catalog (OSC, Guillochon et al. 2017), the OGLE survey portal¹⁷ (Udalski 2003; Udalski et al. 2015), and the ASAS-SN portal¹⁸ (Kochanek et al. 2017; Jayasinghe et al. 2019), all of which are also integrated into our `Viewer`.

So far, the `Viewer` enabled all the ZTF-based results reported by the SNAD team (Malanchev et al. 2021; Aleo et al. 2022; Pruzhinskaya et al. 2022), whose summary can be found in the SNAD catalog,¹⁹ currently hosting 144 candidate transients found by the SNAD team on ZTF DRs, while being absent from systematic searches from other groups.

¹² <https://snad.space>

¹³ <https://www.ztf.caltech.edu/ztf-public-releases.html>

¹⁴ <https://www.lsst.org>

¹⁵ <https://irsa.ipac.caltech.edu>

¹⁶ <https://fritz.science>

¹⁷ <http://ogle.astrouw.edu.pl>

¹⁸ <https://asas-sn.osu.edu>

¹⁹ <https://snad.space/catalog/>

Once it was made publicly available, the `Viewer` made its way from being an infrastructure project used exclusively by SNAD experts to becoming a valuable community resource for scrutinizing ZTF DR objects. It is currently being integrated into two ZTF brokers, ANTARES,²⁰ (Matheson et al. 2021), and Fink²¹ (Möller et al. 2021) and also in the Young Supernova Experiment marshal (Jones et al. 2021; Coulter et al. 2022). It also counts on average, a few dozen unique visitors per day from multiple countries. Beyond ZTF DRs, it provides access to the ZTF alert photometry and light curves from Pan-STARRS (Flewelling et al. 2020) and Gaia (Gaia Collaboration et al. 2016, 2022) surveys. It is an ideal entry point for the photometric investigation of various types of variable objects, including active galactic nuclei (AGNs), Milky Way variable stars, and microlensing events.

In what follows, we give details on the currently available services and their underlying implementation. Details about ZTF DRs are given in Section 2. Section 3 notifies users about the displayed information available and the choices which lead to them. Section 4 describes the only part of the `Viewer` that is, for the moment, private, where SNAD experts can annotate individual objects for subsequent internal use. Sections 5 and 6 describe details of our implementation and is directed toward developers and researchers who may find our experience useful in the construction of similar systems. We present our conclusions and plans for further developments in Section 7.

2. The Zwicky Transient Facility Data Releases

ZTF observes all of the visible Northern sky, covering from 25,000 to 30,000 square degrees having a field of view of 47 square degrees in *gri* passbands, and reaching a limiting median *r*-magnitude of ~ 20.6 for a typical exposure of 30 seconds. It runs both public and private surveys with different cadence, exposure, and passband usage. The public survey is the source of the photometric alerts sent to community brokers, including ALerCE²² (Förster et al. 2021), AMPEL²³ (Nordin et al. 2019), ANTARES (see footnote 21) (Matheson et al. 2021), Fink (see footnote 22) (Möller et al. 2021), Lasair²⁴ and MARS.²⁵ It operates using a differential photometry pipeline that triggers alerts using current observations and the ZTF reference catalogs. The survey has had two phases so far. During Phase I (2018 March–2020 September) it had a 3 days cadence for extra-galactic fields and a 1 day cadence for Galactic fields, while in Phase II (from 2020 December onwards) it switched to a homogeneous 2 days cadence all over the observable sky.

²⁰ <https://antares.noirlab.edu>

²¹ <https://fink-broker.org>

²² <https://alerce.science>

²³ <https://github.com/AmpelProject>

²⁴ <https://lasair-ztf.lsst.ac.uk>

²⁵ <https://Mars.lco.global>

The public survey, which is openly available, uses 30 s exposures and primarily operates in *gr* passbands. The private survey used 60% of its observational time during Phase I and 50% during Phase II. It includes a significantly higher fraction of *i*-passband observations and also high-cadence data reaching hundreds of consecutive observations of the same field during a single night.

ZTF DRs were announced every six months from DR 1 (2019 May) to DR 4 and then the releases switched to a bimonthly schedule. Each DR covers observations for both private (from the start of the survey up to 18 months before the release date due to a proprietary period) and public surveys (from the start of the survey up to a few weeks before the release date). The ZTF DR photometric pipeline is different from the alert pipeline—it is based on source extraction for individual frames and subsequent cross-matching of these sources along all frames within a single pair of observation field/CCD quadrant and passband. This leads to two peculiarities of the data-release objects compared to the alert stream: (1) object light curves do not include non-detections,²⁶ (2) a single sky source can be represented by more than a *dozen* ZTF DR objects due to three passbands and overlapping observation fields.

There are other important differences between the DRs and the alert stream: for example, source identifiers are independent and have different formats, the data-release pipeline does not contain the bogus-to-real image classification step which is a part of the alert production process (Duev et al. 2019), DRs provide Heliocentric MJD (HMJD) time of the middle of exposure while alerts use exposure start JD, among others. Alerts, in turn, are based on source detection via difference imaging, which is more effective for extragalactic sources. For example, most of the ZTF Bright Transient Survey supernovae (Fremling et al. 2020; Perley et al. 2020) are missing from the DRs. However, some robust SN candidates not presented in the alert stream were found in ZTF DRs by the SNAD team in Aleo et al. (2022), Pruzhinskaya et al. (2022). Moreover, the DR light curves have better limiting magnitudes and may also include observations that do not pass the difference imaging pipeline detection threshold (see Section 3.4.2 and Figure 4). To address these issues IPAC provides a service for PSF-fit forced-photometry,²⁷ which has both deeper limiting magnitudes and includes non-detections, but can only be used for individual sky positions.

IRSA IPAC (see footnote 16) provides different ways to access ZTF DRs: through the web interface, light-curve API calls, and via bulk-downloadable files. Since the first two options are not scalable to access the dozens of millions of light curves needed for the SNAD anomaly detection pipelines, we decided to use bulk-downloadable files, which we converted to our internal database

²⁶ The alert stream has a 5σ detection threshold, but it also provides historical measurements up to 30 days before each detection.

²⁷ <https://web.ipac.caltech.edu/staff/fmasci/ztf/forcedphot.pdf>



Figure 1. SNAD Viewer homepage, version 2022.11.2.

format (further described in Section 5.1). However, this option does not provide as much data as the other two interfaces, limiting object properties to: average coordinates, field, readout-channel identifiers, and passband. Similarly, detection properties are restricted to Heliocentric MJD of the middle of the exposure, the magnitude and its uncertainty, the color correction coefficient, and the quality flags.

It is difficult to overstate the importance of ZTF DR for time-domain astronomy. It has already motivated searches and studies of AGNs (e.g., Sánchez-Sáez et al. 2021), strongly lensed QSOs (e.g., Stern et al. 2021), microlensing (e.g., Rodriguez et al. 2022), variable stars (e.g., Chen et al. 2020; Kupfer et al. 2021), young stellar objects (e.g., Kuhn et al. 2021), eclipsing binary systems (e.g., Kosakowski et al. 2022) and, unexpectedly, supernova-like transients (Pruzhinskaya et al. 2022). Moreover, their volume and complexity, combined with their timely existence as a precursor to LSST, makes ZTF DR a unique ground for preparing data mining and machine learning techniques (Malanchev et al. 2021; Aleo et al. 2022) which will be of paramount importance for the next generation of telescopes and surveys. We did our best to make the Viewer a convenient place to inspect photometric and complementary ZTF DR data, a web portal that can serve all researchers interested in deep investigation of individual objects.

3. The SNAD ZTF Viewer Web-portal

3.1. Homepage

The Viewer homepage, shown in Figure 1, gives a brief description of the website and references to the data sources available through the portal. Its header contains a login link

(see Section 4 for SNAD internal resources description) and two search fields: one for ZTF object identifier (OID) and another one for cone search. The OID search field supports 15–16 digit strings for the ZTF DR identifier and 3-digit strings for the SNAD catalog objects (see footnote 20). The cone search field accepts an equatorial coordinate string in various formats²⁸ as well as object identifiers which are resolved to sky positions using SIMBAD (Wenger et al. 2000). The cone search radius is to be specified in arcsecond—the default value is 1" and the maximum supported radius currently is 60".

3.2. Cone-search Results Page

After the user requests a cone search, they are redirected to a page displaying the table of search results. This page has three states: (1) requested string cannot be parsed or resolved via SIMBAD, (2) no results are found, or (3) the search is successful and a table of matches is shown. For the last case, resolved coordinates of the sky position are shown as well as all matched ZTF OIDs. An example is shown in Figure 2. We note that no crossmatch to ZTF objects is performed at this step. Thus, due to the ZTF DR object definition, the user may see multiple object identifiers corresponding to a single sky source. After selecting one of the results, the user is redirected to the object page.

²⁸ The `astropy.coordinates.SkyCoord` class resolves the coordinates, thus supporting most common formats of coordinate representations (Astropy Collaboration et al. 2022).

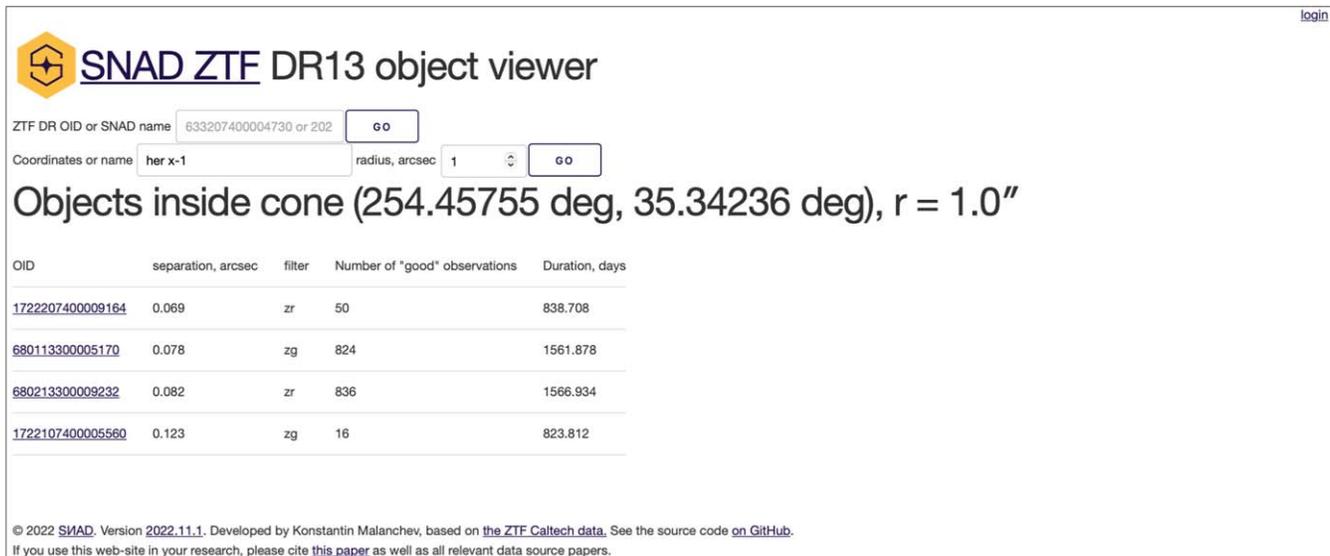


Figure 2. Cone-search result page for a request “her x-1” and 1” cone radius.

3.3. ZTF Object Page

Figure 3 shows the upper part of the object page, the lower part contains external (non-ZTF) catalog cross-matches (see Section 3.6), light-curve features (see Section 3.7) and the ZTF DR photometry table. The ZTF object page is built from multiple blocks, and we describe the most important ones in the next sections.

3.4. Light-curve Plot

The light-curve plot is the main element of the object page. By default, it shows ZTF DR light curves for the given object and all neighboring objects found within the cone of the default search radius (1”). Data points from the target object have larger markers, so that it is easier to distinguish them visually from those of nearby objects. This representation is configurable, and the user can hide light curves of arbitrary objects and change the search radius. It is also possible to display a “short” light curve, restricted to dates corresponding to those of the available private survey. This allows visualization of a homogeneous cadence and overall survey properties presented in ZTF observations. The SNAD team used this functionality in Malanchev et al. (2021), Pruzhinskaya et al. (2022), and Aleo et al. (2022).

A period-folded representation of light curves is also available, see an example in Figure 6. By default, it uses a period corresponding to the highest Lomb–Scargle periodogram peak (Lomb 1976; Scargle 1982) calculated for the target ZTF DR object (Section 3.7) and zero phase with respect to HJD = 58000. The user can change both phase and period,

which can possibly be obtained from multiple sources listed in the “Summary” section just below the plot (see Section 3.8).

Both the original and the period-folded light curves can be downloaded in PNG or PDF formats, generated using the `matplotlib` library (Hunter 2007). These plots display the ZTF object neighbors and the ZTF range and can also handle user-configured parameters such as custom titles and additional photometric data.

Another available customization for the representation of photometric data is the choice between magnitude and flux and between their apparent and difference values. Since ZTF DRs provide the photometric data in magnitudes, we transformed them into fluxes using the AB-system zero-point,

$$f = 10^{-0.4(m-8.9)} \text{ Jy}, \quad (1)$$

$$\sigma_f = 0.4 \ln(10) \cdot f \cdot \sigma_m, \quad (2)$$

where m and σ_m are the apparent magnitude and its uncertainty as reported in the ZTF DR, f and σ_f is our estimate of the apparent spectral flux density and its uncertainty in janskies. The ZTF reference catalog (see Section 6.4 for the technical details) provides the reference magnitude and corresponding uncertainty for each OID. When a user selects the differential photometry option, we load these values for each object and use them for differential flux density estimation,

$$\Delta f = f - f_{\text{ref}}, \quad (3)$$

$$\sigma_{\Delta f} = \sqrt{\sigma_f^2 + \sigma_{f_{\text{ref}}}^2}, \quad (4)$$

where f_{ref} and $\sigma_{f_{\text{ref}}}$ are the reference flux density and its uncertainty determined by Equations (1) and (2). Difference



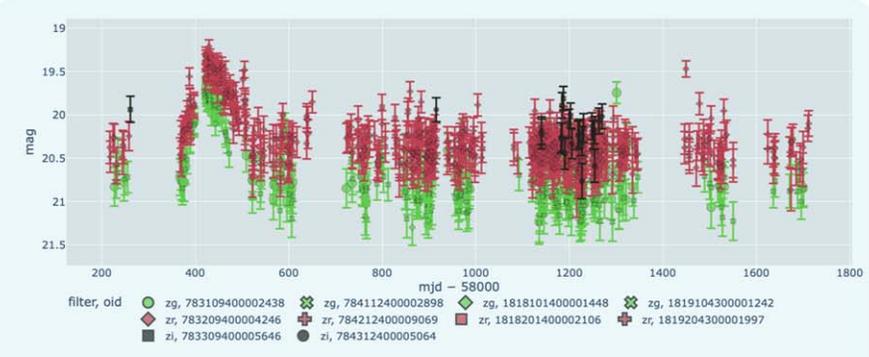
SNAD ZTF DR13 object viewer

login

ZTF DR OID or SNAD name:

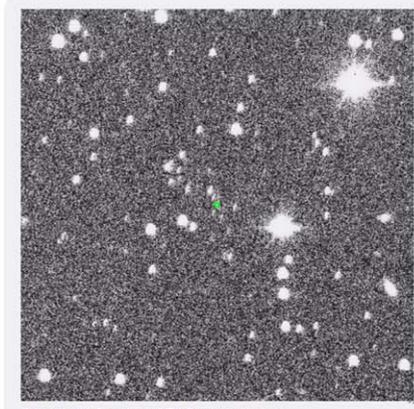
Coordinates or name: radius, arcsec:

SNAD183 — 783109400002438



Download [PNG](#), [PDF](#), [CSV](#)
 "Short" light curve: 58194.0 ≤ MJD ≤ 59280.0
 Full light curve Folded light curve
 Closest Antares object, diff-photometry Closest Pan-STARRS object, apparent Closest Gaia object, apparent
 Magnitude Flux diff Magnitude diff Flux

Section 3.4



Open in JS9 Download FITS Product directory

Section 3.5

Summary

Name: ZTF18acewtzj (0.403" [Alerce](#)), ZTF18acewtzj (0.481" [Fink](#)), PSO J121.0257+55.2658 (0.320" [Pan-STARRS DR2 Stacked](#)), AT2018mbp (0.011" [Transient Name Server](#))

Period, days: 1.000 ([periodogram](#) S/N=26.979)

Average mag (including neighbourhood): zg 20.65, zr 20.25, zi 20.18, (zg-zr) 0.40

Extinction: SFD E(B-V) = 0.05

Search in brokers: [ALeRCE](#), [Antares](#), [Fink](#), [MARS](#)

Coordinates: Eq 121.02564 55.26591, Gal 162.705 32.1966

Section 3.8

Neighbours

Different passband, same field

search radius, arcsec
[783209400004246](#) (0.049"), [783309400005646](#) (0.132")

Different field

search radius, arcsec
[784112400002898](#) (0.126"), [1818101400001448](#) (0.132"),
[784212400009069](#) (0.137"), [1818201400002106](#) (0.137"),
[784312400005064](#) (0.179"), [1819104300001242](#) (0.237"),
[1819204300001997](#) (0.237")

Metadata

nobs: 217	duration: 1473.019	ref_mag: 20.492
ngoodobs: 145	fieldid: 783	ref_magerr: 0.046
filter: zg	rcid: 35	ref_flags: 0
coord_string: 121.02564 55.26591		

Sections 3.6 & 3.7



Figure 3. The upper part of the object page for SN candidate (Pruzhinskaya et al. 2022) OID 783109400002438/SNAD183/AT2018mbp. Light-curve plot (top left) shows object’s *g*-passband photometry by large green circles, while the smaller symbols show photometry of eight more ZTF DR objects found within one arcsecond from its position in *gri* passbands and four overlapping ZTF fields. The horizontal axis label, “mjd,” denotes Heliocentric MJD at the middle of the exposure, as reported in ZTF DR. The FITS viewer (top right) shows a scientific image for a detection near the peak selected by a user. Aladin (Bonnarel et al. 2000; Boch & Fernique 2014) Sky Atlas (below the FITS image) shows the corresponding image from Pan-STARRS (Flewelling et al. 2020) while the small blue circle indicates the object position.

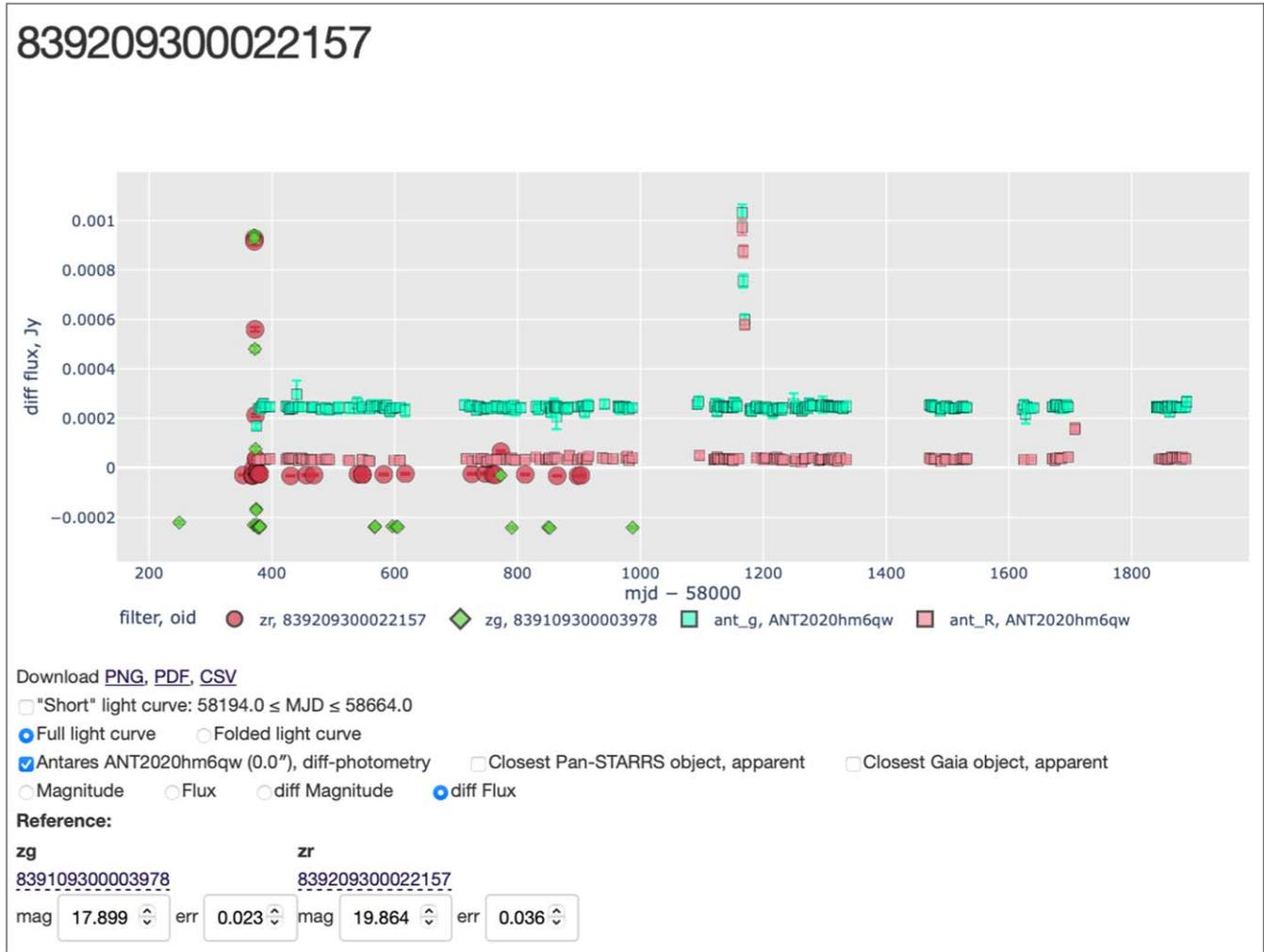


Figure 4. Light curve of a cataclysmic variable (Blagorodnova 2017) AT2017hue/ZTF18abvtemh/ZTF DR 839209300022157 in difference fluxes. Alert data (light squares showing the outburst around MJD 59200) alone gives only a limited picture of an object's variability. Here, with ZTF DR4 (dark circles and diamonds with the outburst around MJD 58400) observations, it can be seen that the object is recurrent and has already reached the same level of the brightness before. The bottom part of the screenshot shows the reference magnitude selection block.

magnitudes are considered to have asymmetric uncertainties,

$$\Delta m = 8.9 - 2.5 \lg \Delta f, \quad (5)$$

$$\sigma_{\Delta m}^+ = -2.5 \lg \left(1 - \frac{\sigma_{\Delta f}}{\Delta f} \right), \quad (6)$$

$$\sigma_{\Delta m}^- = -2.5 \lg \left(1 + \frac{\sigma_{\Delta f}}{\Delta f} \right). \quad (7)$$

The user can change the reference values, which can be useful for cases where the ZTF reference catalog gives unsuitable reference magnitudes, e.g., when a reference image contains

the considered object around its maximum light. An example of a difference flux density plot is shown in Figure 4.

3.4.1. External Light Curves

Currently, three external photometry data sources are supported: the ANTARES (see footnote 21) broker for ZTF alert data, Pan-STARRS DR2²⁹ and Gaia DR3.³⁰ The light-curve plot can include additional epochs from all of them. We used the corresponding APIs to perform a 5" cone search

²⁹ <https://outerspace.stsci.edu/display/PANSTARRS/>

³⁰ <https://www.cosmos.esa.int/web/gaia/dr3>

centered on the ZTF object position and choose the closest object to be included in the `Viewer` light curve.

3.4.2. ZTF Alert Light Curve

Apart from the availability of non-detections, ZTF alert pipeline is intrinsically different from those in the DR (see Section 2 and Bellm et al. 2019b for details). Moreover, the alert stream delivered to the ZTF brokers may contain more recent observations. Therefore showing this data together with the ZTF DR light curves might bring important insights on the astronomical objects inspected with the `Viewer`. Moreover, ZTF alert users can find the ZTF DR useful to distinguish, for instance, a supernova from an AGN or a cataclysmic variable, which could have similar alert light curves, but different DR light curves (see Figure 4 for a real-life example of such a case). For this purpose we use the ANTARES (see footnote 21) broker (Matheson et al. 2021) API.³¹

3.4.3. Pan-STARRS DR2

The `Viewer` also uses data from Pan-STARRS. ZTF uses photometric filters with passbands close to the Pan-STARRS filters, and also uses its data for photometric calibration. Moreover, both surveys operate in the northern hemisphere and cover roughly the same part of the sky. Finally, Pan-STARRS DR2 includes time-resolved photometry data in *grizy* passbands from a period before the start of ZTF: from 2010 to 2014 (Flewelling et al. 2020). All these make Pan-STARRS DR2 light curves complementary to ZTF data. Therefore we provide Pan-STARRS DR2 PSF photometry, and we use the MAST API³² for data access.

Pan-STARRS DR2 gives calibrated flux densities in janskies, which we convert into AB-magnitudes using the inverse of Equations (1) and (2),

$$m = 8.9 - 2.5 \lg f, \quad (8)$$

$$\sigma_m = \frac{2.5}{\ln 10} \frac{\sigma_f}{f}. \quad (9)$$

Pan-STARRS provides observation times according to the international atomic time standard at the midpoint of each observation. We transformed this value to Heliocentric MJD in UTC. An example of a combined ZTF-Pan-STARRS light curve is shown in Figure 5.

3.4.4. Gaia DR3

The third data release of Gaia contains epoch photometry in *G*, *BP* and *RP* passbands covering observations between 2014 July 25 and 2017 May 28 (Eyer et al. 2022; Gaia Collaboration et al. 2022).

The Gaia DataLink service³³ (Dowler et al. 2015; Gonzalez-Núñez et al. 2019) provides access to epoch instrument fluxes, f_{Gaia} . We used instrumental zero-points, z_p , and their uncertainties, σ_{z_p} , to convert fluxes to AB-magnitudes (Riello et al. 2021),

$$m = z_p - 2.5 \lg(f_{\text{Gaia}}), \quad (10)$$

$$\sigma_m = \sqrt{\left(\frac{2.5}{\ln 10} \frac{\sigma_{f_{\text{Gaia}}}}{f_{\text{Gaia}}} \right)^2 + \sigma_{z_p}^2}. \quad (11)$$

Flux values and their uncertainties, in janskies, were then obtained using Equations (1) and (2).

Gaia uses the TCB time standard and provides the average transit Barycentric time. We converted it to UTC standard and assumed that for the purposes of the majority of the `Viewer` users, the few seconds-level difference between the Barycentric time and the Heliocentric time (e.g., Eastman et al. 2010) as given in ZTF DR, would not be considered significant. An example of a combined ZTF–Gaia light curve is shown in Figure 6.

3.5. ZTF FITS

We added an embedded JS9³⁴ (Mandel & Vikhlinin 2022) FITS image widget to the right of the light-curve plot (see Figure 3). The user can click on any ZTF DR photometry point to load the scientific image corresponding to that observation. The image is centered on the object position, marked by a green dot, and the default zoom factor is equal to unity, which means that a single screen point corresponds to a single CCD pixel. While we used the JS9 widget without the toolbar and the menu, the link to the fully functional JS9 viewer is provided just below the image itself, in case the user requires additional functionalities.

We used the IRSA IPAC archive³⁵ as the original source of these FITS images. Additionally, we employed our own proxy to work around the problem of loading data into the JS9 widget from a location not allowing cross-origin resource sharing (CORS)³⁶ due to safety concerns (see Section 6.4 for our implementation details). The archive uses the following path encoding for scientific FITS images: YYYY/MMDD/MJDFRA/ztf_YYYYMMDDMJDFRA_FIELDN_PB_c_CN_o_qQ_s-ciimg.fits, where YYYY, MM, DD are the year, month, and day of observation in UTC, MJDFRA is the fractional part of the MJD at the start of exposure, FIELDN is the ZTF field number, PB is the ZTF passband name (zg, zr or zi), CN is the CCD identifier (01 to 16), and Q is the CCD quadrant identifier (1 to 4). A typical example of a path for a source is: 2019/0314/432616/ztf_20190314432616_000633_zr_c07_o_q4_sciimg.fits.

³³ <https://gea.esac.esa.int/data-server/datalink>

³⁴ <https://js9.si.edu>

³⁵ <https://irsa.ipac.caltech.edu/ibe/data/ztf/products/sci/>

³⁶ <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

³¹ <https://nsf-noirlab.gitlab.io/cscd/antares/client/index.html>

³² <https://catalogs.mast.stsci.edu/panstarrs/>

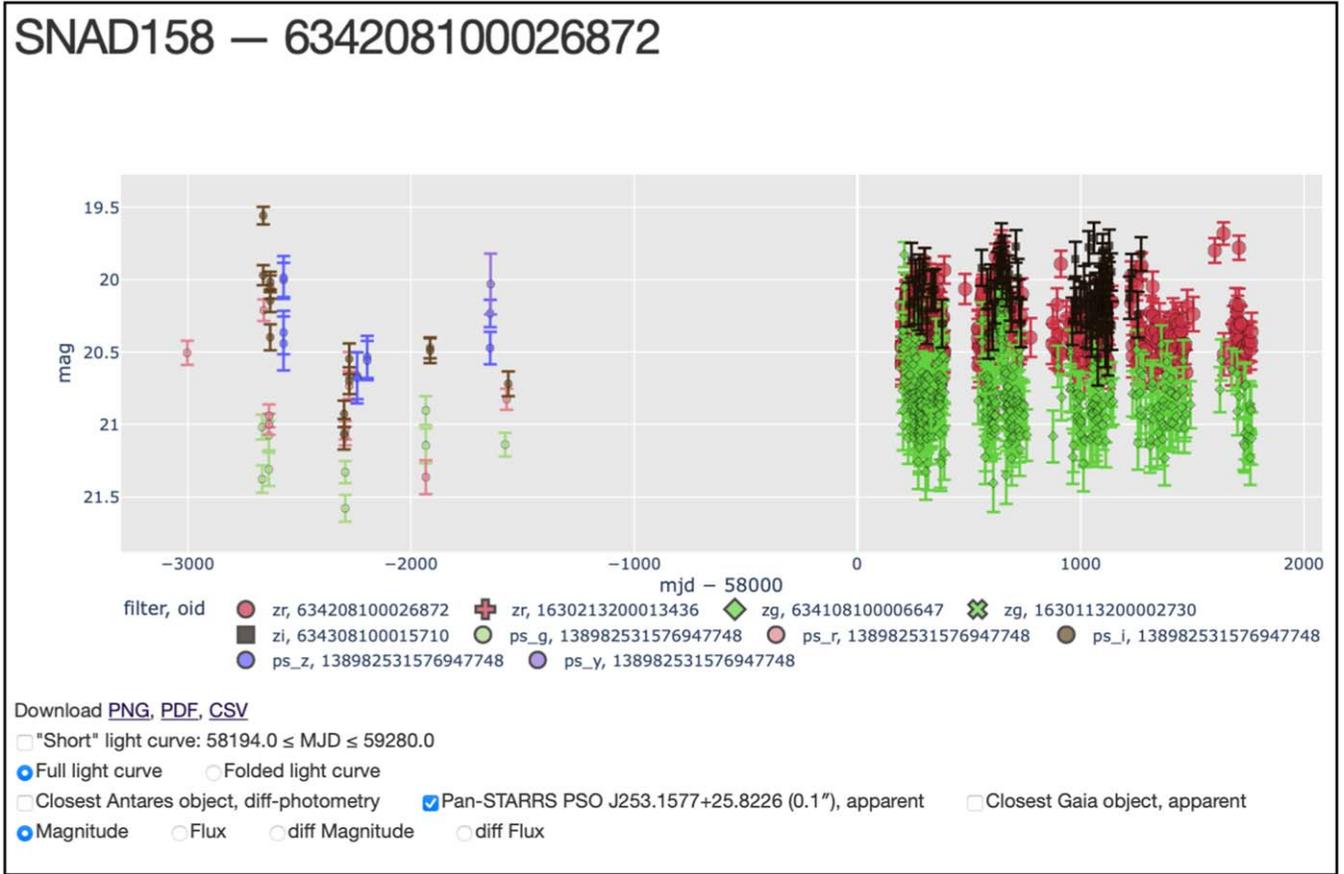


Figure 5. ZTF DR13 and Pan-STARRS DR2 light curves of AGN candidate (Aleo et al. 2022) SNAD158/AT2018lzd/ZTF DR OID 634208100026872/PSO J253.1577+25.8226.

The main issue we faced while assembling this path is that the ZTF DR bulk-downloadable files (Section 2) do not provide enough information to allow for their unambiguous reconstruction. They provide only the HMJD and the read-out channel identifier `rcid` (0 to 63). The latter is sufficient to reconstruct the CCD and quadrant identifiers,

$$\text{CN} = \left\lfloor \frac{\text{rcid}}{4} \right\rfloor + 1, \quad (12)$$

$$Q = \text{rcid} \bmod 4 + 1. \quad (13)$$

The more challenging part is reconstructing the MJD of the beginning of the exposure from the HMJD in the middle of it, without information about the exposure duration and having a precision of around a second (HMJD is rounded to five decimals). We solved this issue by getting an estimate of the required `MJDFRA`³⁷ and by identifying the closest available value in the archive, which resulted in a working approximation for the FITS file path.

³⁷ We used `astropy.time.Time.light_travel_time` method to get the difference between MJD and HMJD.

3.6. External Catalogs Cross-matching

One of the main purposes of the Viewer is to give experts relevant information about a given source beyond what is available within ZTF DR photometric data. We achieved this by providing cross-matched information from multiple catalogs, including: the General Catalog of Variable Stars (GCVS, Samus' et al. 2003, 2017), the International Variable Star Index (AAVSO VXS, Watson et al. 2006), the Atlas Catalog of Variable Stars (ATLAS-VAR, Heinze et al. 2018), the Sloan Digital Sky Survey DR16 Quasar Catalog (Lyke et al. 2020), the ZTF Catalog of Periodic Variable Stars (Chen et al. 2020), the Spitzer/IRAC Candidate YSO Catalog (SPICY, Kuhn et al. 2021), Pan-STARRS DR2 (Flewelling et al. 2020) (stacked photometry table), the Transient Name Server³⁸ (TNS), the Open Supernova Catalog (OSC, Guillochon et al. 2017), the OGLE III catalog of Variable Stars (Udalski 2003), the SIMBAD astronomical database (Wenger et al. 2000), the Gaia EDR3 Distance Catalog (Bailer-Jones et al. 2021) and the

³⁸ <https://www.wis-tns.org>

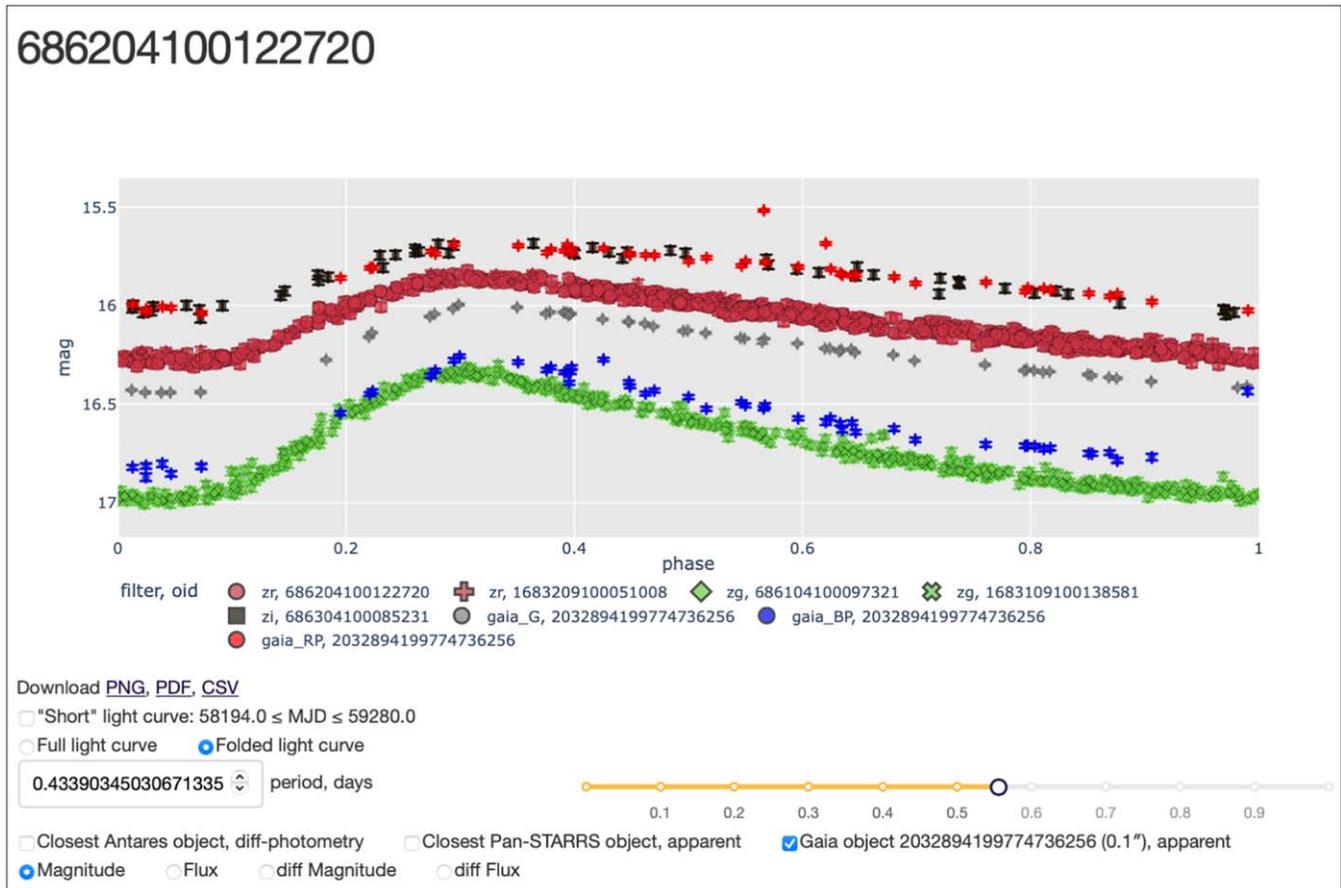


Figure 6. ZTF DR13 and Gaia DR3 period-folded light curves of RR Lyrae variable (Clementini et al. 2019) ZTF DR OID 686204100122720/Gaia DR3 Source ID 2032894199774736256. The period is estimated automatically, while the non-zero phase is selected by the user.

Gaia DR3 (Gaia Collaboration et al. 2022). For most of the listed catalogs, we used the VizieR service³⁹ (Ochsenbein et al. 2000), MAST⁴⁰ for Pan-STARRS, Open Astronomy Catalog API⁴¹ for the OSC, the ESA Gaia Archive,⁴² and our own services for the remaining catalogs (see Section 6.2 for the implementation details). For each considered catalog, the Viewer displays only a few selected columns, while also providing a link to the full record at the external resource. It also provides the cross-match result for the default cone radius but gives the user the ability to set a different value.

3.7. Light-curve Features

We created the Viewer for the analysis of objects suggested by machine-learning algorithms which use time-series features extracted from ZTF photometrical data. SNAD team uses the

³⁹ <https://vizier.u-strasbg.fr>

⁴⁰ <https://archive.stsci.edu>

⁴¹ <https://astrocats.space>

⁴² <https://gea.esac.esa.int/data-server/datalink>

light-curve⁴³ (Malanchev 2021; Malanchev et al. 2021) feature extraction toolkit to prepare the input for these algorithms. Thus, we provide a list of extracted features on the object page of the Viewer, allowing the user to choose between different versions—each set corresponds either to a SNAD paper or a major version of the light-curve package. We also use the highest periodogram peak as the default period for the folded light-curve plot (see Section 3.4) and in the Summary section (see Section 3.8).

3.8. Summary Section

The summary section is an essential part of the object page because it gives the expert a succinct overview of all the data available regarding the chosen source. Important parts of this section are:

1. Period: given by light-curve features (see Section 3.7) and variable star catalogs.

⁴³ <https://github.com/light-curve>

2. Distance: collected from all data sources providing a distance or a redshift estimate, the most valuable catalogs for this purpose are Gaia EDR3 distances (Bailer-Jones et al. 2021), OSC (Guillochon et al. 2017), and TNS.
3. We provide two estimates of Extinction: from the SFD 2-D map (Schlegel et al. 1998; Schlafly & Finkbeiner 2011) and the Bayestar 3D map (Green et al. 2015, 2019), if Gaia EDR3 distance is available. We used the `dustmaps` Python package (Green 2018) in both cases.
4. Possible object types are collected from all available data sources including catalogs like AAVSO VSX, and live services like TNS.
5. We also give ML classifications separately to represent probabilistic classification given by ZTF brokers Alerce and Fink, as well as by some catalogs like Gaia DR3.
6. Average magnitudes and $g - r$ color are calculated over the current object as well as all selected ZTF neighboring objects.
7. Search in ZTF brokers: links for cone searches in Alerce, ANTARES, Fink and Mars.

The summary section has a dynamic layout, and therefore it does not show “missing” values. For instance, the distance and type parts are hidden automatically if no corresponding data are available. Also, the summary section changes its content on-the-fly if a user changes sections it depends on, e.g., the cone-search radius for some external catalog (see Section 3.6) or light-curve version (see Section 3.7).

4. The Knowledge Database

Beyond the public user-interface features described above, the `Viewer` has a private section called the Knowledge Database (KDB), which is filled and used by authorized experts of the SNAD team. The KDB was built to allow SNAD experts to share expertise among themselves. Whenever a given candidate is presented to an expert, it is followed by extensive analysis based on visual screening of all the photometric data in the `Viewer`, literature review, photometric model fitting, or follow-up observations. All this information is combined in a final judgment about the nature of the candidate, which is recorded by the expert in the KDB.

After login, the expert is directed to an enhanced version of the object page, which contains one more block representing the KDB record for the current OID (Figure 7). The block displays the most recent classification and the description record for the object, as well as a history of all previous records, including its date and author. The expert can also provide a description of the object in free text format, mainly used to clarify classification choices and extra data sources.

The classification is represented in terms of a non-hierarchical tag system so that each object can have multiple tags. The tag itself is a short string that shows an associated description at pointer hover. Experts can change the order tags and add new ones by accessing the tag editor page. Since the

SNAD team works as one coherent group of specialists, every user has the power to change the tag list, which is available to all users. However, the same structure can be personalized for other projects or collaborations, allowing each user to see their own set of chosen tags if necessary.

Currently, the KDB contains more than 50 tags which can be split into the following categories: types and sub-types of variability such as “Eclipsing,” “EA,” “EB,” and “EW;”⁴⁴ image and photometric pipeline artifacts like “artifact” and “defocusing;” properties of classification such as “uncertain” and “non-cataloged” which are only meaningful when combined with other tags; and cross-team communication tags like “TNS_candidate,” which states the object is worth submitting to TNS. A detailed description of the currently available tags was reported by Pruzhinskaya et al. (2022).

Figure 8 shows the KDB table page with an interactive table of all tagged objects. An expert can apply filters to individual columns, for example, to see all objects marked with a specific tag or all records modified by someone. The KDB also has programmatic access, which can be used for more sophisticated analysis. We further describe the implementation details of the KDB API service in Section 6.5.

In the context of the adaptive learning algorithms developed by the SNAD team, the final state of a specific learning model is a direct consequence of the feedback provided by the expert during training. Thus, the historical record stored in the KDB is crucial to allow for reproducibility of results, as well as to isolate probable causes for divergence in models trained by different experts. The historical record also allows experts to immediately access each other’s judgments, which has proven efficient, especially when experts come from different scientific backgrounds or train models with different goals. Finally, the SNAD team is currently developing machine learning models which aim at using tags from the KDB as prior information. This will optimize the allocation of human resources since experts will not need to start from scratch when training a model for a new purpose.

5. Infrastructure

The viewer infrastructure is built from multiple services which mainly communicate via the HTTP protocol (Fielding et al. 1999). We define in this paper a module as a program configured as a single `Docker`⁴⁵ container, a service as a closely connected set of modules that are configured by a single `Docker-Compose` file, and a service instance as a set of containers, volumes, and networks deployed by a `docker-compose` tool. Since the `Viewer` does not have high availability requirements, most of our services have only one instance running on a virtual private server (VPS). However,

⁴⁴ We adopted types of variability from the GCVS (Samus’ et al. 2017), and VSX (Watson et al. 2006): “Eclipsing” is used for any eclipsing binaries, while “EA,” “EB” and “EW” are sub-types of eclipsing binaries.

⁴⁵ <https://docker.com>

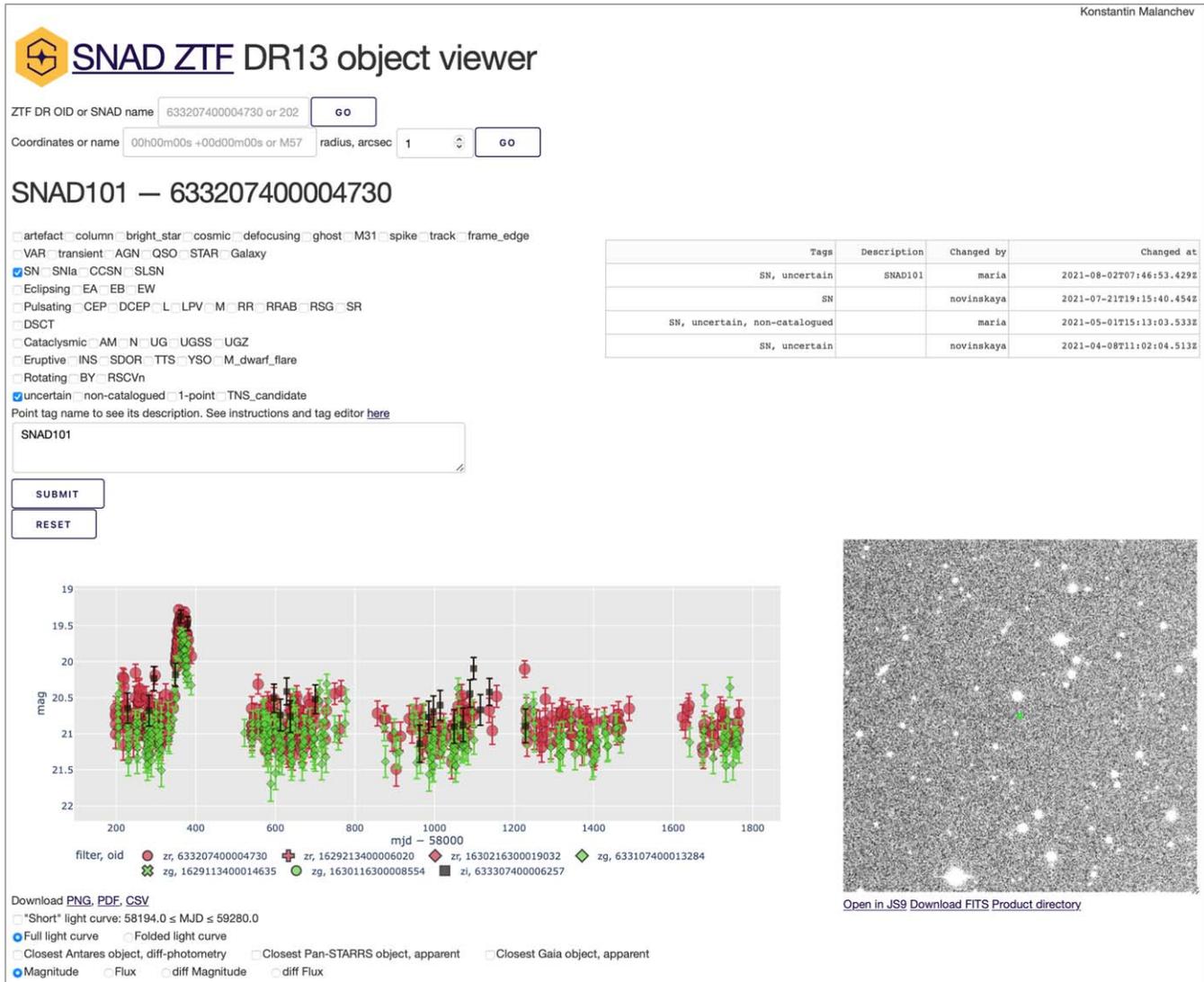


Figure 7. The upper part of the object page contains the SNAD Knowledge Data Base. It represents SN candidate (Pruzhinskaya et al. 2022) ZTF DR OID 633207400004730/SNAD101/AT2018lwh. According to this page, the most recent classification of this object is supernova (tag “SN”) without obvious sub-type (no tags like “SN Ia”), the object does not possess a recorded classification in any of the public catalogs checked by the expert (tag “uncertain”). The top right table shows that this object was independently inspected by SNAD experts four times.

the ZTF DR database API (see Section 6.2.1) and the ZTF FITS caching proxy (see Section 6.4) services have significant hardware requirements which do not meet our VPS budget. Therefore we have instances running those services at dedicated servers located at two academic institutions (currently at the Sternberg Astronomical Institute and at the University of California, Irvine). Individually, each of these dedicated servers has lower availability than the VPS due to possible network, power, and hardware issues. Nevertheless, we duplicated the services in those two services, increasing the availability at the system level.

The choice of the multi-service architecture behind the Viewer design is based not only on budget limits and availability requirements but also on the need for low maintainability effort and easy deployment. Most of the services are self-consistent and require no persistent data for deployment to a different host, which is a desirable feature whenever the host server needs to be replaced. This is also an advantage in case we decide to perform additional on-demand deployments in the future, dispatching cloud instances automatically for instance. Issue localization and debugging are also easier in a multi-service architecture because, generally, in the case of unexpected problems, a service will fast-fall,

OID	Tags	Description	Changed by	Changed at
filter data...	AGN	SNAD1		
634108100006647	AGN, SN, uncertain	SNAD158	maria	2021-10-21T22:22:11.362000+00:00
643105300009229	AGN, SN, uncertain	SNAD153	maria	2021-10-21T21:39:27.291000+00:00
716211100008498	AGN, SN, uncertain	SNAD185	maria	2022-07-21T10:00:49.987471+00:00
718111400007010	AGN, SLSN, uncertain	SNAD187	maria	2022-10-19T12:35:36.467697+00:00
718205300006523	AGN, uncertain, non-catalogued	SNAD155	maria	2021-10-22T09:30:51.131000+00:00
718211400012193	AGN, SLSN, uncertain	SNAD187 AGN according to Gaia	maria	2022-08-01T15:32:37.408774+00:00
719202100004008	AGN, SN, uncertain	SNAD154	maria	2021-10-21T22:06:23.389000+00:00
756202100006670	AGN, SN, uncertain	SNAD146	maria	2022-07-13T19:04:58.201760+00:00
758205100001118	AGN, SN, uncertain	SNAD103, weak SN candidate	maria	2022-07-20T14:18:47.505509+00:00
758209200010983	AGN, SN, uncertain	SNAD105	maria	2021-08-02T11:13:34.573000+00:00
790213400013270	AGN, SN, uncertain	SNAD110	maria	2022-07-20T13:49:44.253114+00:00
824204400004814	VAR, AGN, non-catalogued	SNAD148	maria	2021-10-21T08:22:33.632000+00:00

Figure 8. KDB table page. An expert has applied “AGN” tag filter and “SNAD1” description filter to show AGN candidates from the SNAD catalog.

making it easy to identify which service is not available at a given moment. The same reasoning applies to our choice of having one database management system (DBMS) module per service instead of a single centralized DBMS service holding multiple databases: (1) with a single DBMS instance, any outage or network issue would cause an interruption of all services relying on it, and (2) additional network configuration would be required to help services discover and secure access to this DBMS instance, while, in our approach, `Docker compose` solves this problem via virtual networks connecting each API module to its DBMS companion. The `Viewer` service itself is not sensitive to the degradation of other services; for example, if one of our catalog services goes down, the `Viewer` shows that this catalog is not available, but otherwise, it continues to operate as usual. The multi-service approach also allows the use of individual services by multiple projects; for instance, the SNAD collaboration accesses the `Viewer` APIs via Python scripts and notebooks.

In our architecture, individual services are configured as a set of `Docker` containers bundled to have a common private virtual network and volumes using `Docker Compose`. Each of the servers we use has a common infrastructure configured as a dedicated `Docker Compose` and includes the following modules,

1. `nginx-proxy`⁴⁶ is a reverse proxy that gives access to our HTTP services via different domain names (virtual hosting);
2. `acme-companion`⁴⁷ gets and automatically renews TLS certificates via Let’s Encrypt authority,⁴⁸ making it

possible to have secure HTTPS access throughout the system;

3. `dyndns53`⁴⁹ creates and updates type A CNAME records for service domain names via Amazon Web Services’ Route 53 DNS.⁵⁰

The `Viewer` design choices are similar to those the ZTF project team made for their Fritz Astronomy Marshal (see footnote 17). Its infrastructure is also based on multi-service architecture, services are managed by `Docker-Compose` and communicate via HTTP RESTful APIs. The Fritz Marshal also acts as an alert broker via its Kowalski component.⁵¹ Some of the software and framework choices of Fritz and the `Viewer` also match: both use Python as a main programming language and `aiohhttp`⁵² as a framework for APIs. However some design and software choices differ, for instance, Fritz uses a document data model via the MongoDB data management system, while the `Viewer` uses relational model (see Section 5.1). Both approaches have their pros and cons, and as we mention later, our data model choice was primary based on the requirements coming from the SNAD machine learning pipelines, while the document data model would work faster for the needs of the portal.

We show in Figure 9 the infrastructure of the `Viewer`, representing an overview of all the services and the individual modules. The data flow is represented using lines and arrows: the `Viewer` consumes data from many services and external APIs, while experts add records to the Knowledge Database using the `Viewer`. We further describe the implementation details of individual `Viewer` services in Sections 5.1 and 6.

⁴⁶ <https://github.com/nginx-proxy/nginx-proxy>

⁴⁷ <https://github.com/nginx-proxy/acme-companion>

⁴⁸ <https://letsencrypt.org>

⁴⁹ <https://github.com/hombit/dyndns53>

⁵⁰ <https://aws.amazon.com/route53/>

⁵¹ <https://github.com/dmitryduev/kowalski>

⁵² <https://docs.aiohhttp.org>

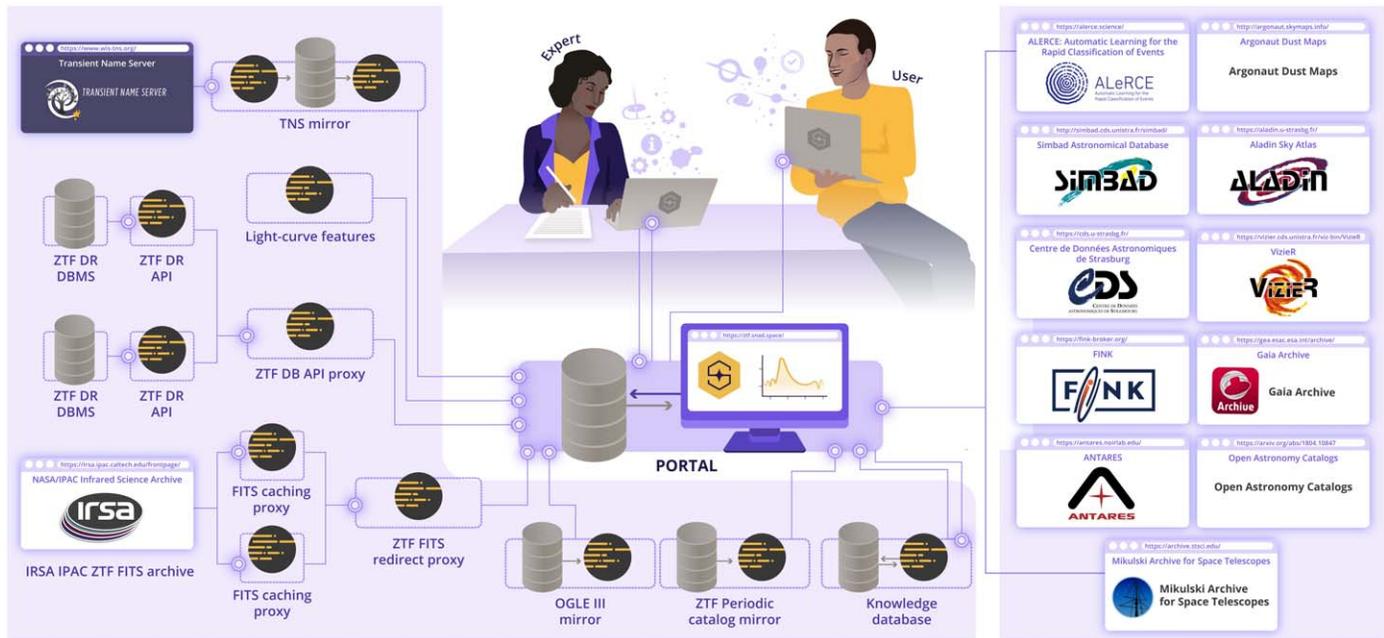


Figure 9. Diagram of the service infrastructure. Dashed rectangles represent individual services, circles are web modules and complimentary scripts, cylinders are database management systems, and the display is the Portal module. Parallelograms are external services used by the Viewer. Lines show data flow, double circles mark data receivers. Arrows show data exchange between modules of a single service.

5.1. Data Storage for ZTF DRs

Since SNAD is focused on the development and application of anomaly detection machine learning algorithms, each of our ZTF DR projects needs to handle millions of light curves containing hundreds of millions of photometric points. Therefore one of our main DBMS requirements was a high performance for analyses which query up to a few percent of volume of a multi-terabyte database. We found that relational columnar databases fit this requirement better than row-based (as PostgreSQL) or document-based (as MongoDB) systems.

Currently the SNAD machine-learning pipeline includes Clickhouse DBMS, which achieved up to 100× higher performance gain over PostgreSQL for our machine-learning needs, while it uses a few times less data storage because of its columnar format and data compression. The Viewer employs the same DBMS setup as we use for our machine-learning pipeline, but it uses a very different query pattern: it queries a single ZTF object or a small number of objects for the cone search (Sections 3.2 and 3.3). This usage pattern is sub-optimal for columnar databases, which translates into delays in the Viewer usage of ZTF DR DB API, due to Clickhouse response times, up to a few hundred milliseconds. However, this solution is robust and fits our computation and storage budget since it does not require an additional copy of the database managed by another DBMS.

We use a simple table schema for each DR, which contains detection and object tables and shares a common object ID

column used for join queries. Cone-searches are performed using pre-computed H3 indices⁵³ (hexagonal hierarchical spherical, Sahr et al. 2003) with a resolution of 10, which corresponds to roughly 2"1 tile edge size. Currently, our Clickhouse database contains ZTF data releases 2, 3, 4, 8, and 13. It has more than a trillion rows and occupies ~15 TB of storage.

5.2. Development

The code development follows a multi-service application approach: the source code of each service is located at a separated Git⁵⁴ repository; we share no code between the different services. The only exceptions are closely related groups of services, namely those coupled with a load-balance service (see Sections 6.2.1, 6.4), which, for convenience, use a repository per group. We use GitHub as a remote Git repository hosting, issue tracker, collaborative tool via its Pull-request functionality, continuous integration and continuous delivery (CI/CD) tool via its GitHub Actions functionality.

We utilize CI/CD for various types of tests: from the automatic unit, integration, and regression tests for our services to manual user interface tests of the Viewer. In order to allow for easy manual testing by all the members of the SNAD development team, we use a GitHub Actions workflow

⁵³ Clickhouse utilizes H3 library <https://h3geo.org>.

⁵⁴ <https://git-scm.com>

which deploys a development version of the `Viewer` for the master `Git` branch and each active `GitHub` pull request. Each such development Portal instance has a domain name <https://pr###.ztf.said.space> with a proper TLS certificate, and runs as a separate `Docker-Compose` project at a VPS for development.

6. Services Implementation

As previously described, the `Viewer` is designed following a multi-service architecture. In this section, we present its most important services.

6.1. The `Viewer` Web-portal

Initially, the `Viewer` was a simple dashboard-like single-page web application for displaying the light curves of ZTF DR1 and a few catalog cross-matches. The only valuable requirements at that moment were development velocity and usage of the Python language at the back-end, which enabled access to various astronomical packages. At that time, we found `plotly`⁵⁵ to be a good solution for interactive graph plotting, allowing us to write both front- and back-ends as a single Python application. We also discovered that the developer of `plotly` had just released the `Dash`⁵⁶ framework, built upon `Plotly.js` and `React.js` JavaScript libraries, which allows adding a Python callback to almost every user-interactable HTML tag and which has a set of useful extensions for interactive data representation. Given that `Dash` fitted our requirements, we decided to implement the `Viewer` in this framework. It is worth mentioning that another useful interactive data visualization framework is available in Python: `bokeh`,⁵⁷ which also provides tools for graphical data but does not cover control over check-boxes, lists, tables, and other elements of the web-page.

`Dash` allowed us to rapidly develop an initial version of the portal with a small number of code lines. However, soon after that, we faced some of its limitations. The main issue with the current `Viewer` implementation is limited opportunities for testing (see Section 5.2 for more details about the development pipeline) because `Dash` forces developers to mix data-model and data-view code. This makes debugging programming issues significantly more challenging and also makes code maintenance more time-consuming. Another specificity of `Dash` is the single-page design of the application, which makes it challenging to properly support a rich URL scheme and per-page specification of the tags inside of `<head>`.⁵⁸ However, `Dash` uses the `Flask`⁵⁹ web framework for HTTP

management, which allowed us to implement a few non-HTML endpoints (for example, for downloadable plots, see Section 3.4) independently from the main application code.

The source code of the web-portal is available via `GitHub`.⁶⁰ The portal would not be possible without the libraries it utilizes, including but not limited to: `astropy` (Astropy Collaboration et al. 2022), `astroquery` (Ginsburg et al. 2019), `dustmaps` (Green 2018), `h5py` (Collette 2013), `healpy` (Zonca et al. 2019), `numpy` (Harris et al. 2020), `pandas` (McKinney 2010; pandas development team 2020), `scipy` (Virtanen et al. 2020).

6.2. Catalog APIs

This section describes the HTTP API services that we built for the databases we host for cross-matching purposes. All services are written in Python and use `Gunicorn` Web Server Gateway Interface (WSGI) HTTP server. Most of the services employ a simple scheme for endpoints, having endpoints like `/api/v1/circle?ra=RA&dec=DEC&radius_arcsec=RADIUS` typically returning data in JSON format.

6.2.1. ZTF DR

The ZTF DR database API has three services: (1) `Clickhouse` DBMS (see Section 5.1), (2) Python API-service which uses a connection to the database and provides HTTP API to access it, and (3) `Nginx`⁶¹ web-server configured to be a fail-over reverse proxy. The first two services have two instances each and they are continuously running at dedicated servers located at different academic institutions (Section 5), while the last one runs at the VPS and proxies queries to the first alive API service instance for higher availability.

The HTTP API service uses the asynchronous Python web framework `aiohhttp`?. Currently, it gives access to two main `Clickhouse` tables for each supported ZTF DR, namely the metadata table and the detection table. We have made our source code available on `GitHub`.⁶²

6.2.2. OGLE III

Since OGLE III (Udalski 2003) provides a user-friendly web interface but does not have a dedicated API, we maintain its mirror. We used the `PostgreSQL`⁶³ DBMS and the `Flask` Python web-framework for the HTTP API implementation. We have made our source code available on `GitHub`.⁶⁴

⁵⁵ <https://plotly.com>

⁵⁶ <https://dash.plotly.com>

⁵⁷ <https://bokeh.org>

⁵⁸ A limited support of the multi-paging was recently introduced by `Dash` version 2.

⁵⁹ <https://flask.palletsprojects.com>

⁶⁰ <https://github.com/snad-space/ztf-viewer>

⁶¹ <https://nginx.com>

⁶² <https://github.com/snad-space/snad-ztf-db>

⁶³ <https://www.postgresql.org>

⁶⁴ <https://github.com/snad-space/snad-ogle3>

6.2.3. ZTF Periodic Catalog of Variable Stars

The set-up for the ZTF periodic catalog of variable stars (Chen et al. 2020) mirror is similar to OGLE III one. This source code is also available on our GitHub.⁶⁵

6.2.4. Transient Name Server

The TNS provides an API that currently implements a 60 s rate limit which is not suitable for our use case, thus also requiring the maintenance of a local mirror from our side. Since TNS is a live service, we need to update its mirror periodically. For this purpose, the TNS mirror service includes not only the PostgreSQL DBMS and the Python aiohttp?? modules but also an additional module that periodically downloads the official TNS daily database dump and ingests it into our database. The source code of this service is also available on GitHub.⁶⁶

6.3. Feature Extraction API

The light-curve feature extraction API wraps a few versions of the light-curve-feature⁶⁷ library as a single module written in Rust with the usage of Rocket⁶⁸ web-framework. The source code of this service is available on GitHub.⁶⁹

6.4. ZTF FITS Caching Proxy

We utilize Nginx as a caching reverse proxy for the IRSA IPAC archive of ZTF FITS files, mainly due to the safety issue with HTTP Cross-Origin Access politics of web-browsers (see Section 3.5). Since we require a client to download the whole FITS image files, whose size is ~ 38 MB each, the available network bandwidth between the client and one of our caching proxies is a limiting factor. Aiming at optimizing both network speed and fault tolerance, we duplicate the Nginx service located on different continents and use geo302⁷⁰ redirecting proxy which sends a 302 “Found” HTTP response containing a URL of the closest alive caching proxy service. The source code is available on GitHub.⁷¹

6.5. Knowledge Database API

The KDB implementation consists of a Python application written with the Django REST framework,⁷² while we use the PostgreSQL DBMS module to hold the underlying data. The REST API has two top-level end-points: one for tags, which have short names, descriptions, and web-page position

indexes, and another endpoint for objects, which are identified by ZTF DR OIDs and have descriptions, sets of tags, authorship, and date of the last change. The API also has basic filtering support which, for instance, allows us to request all objects with a specific tag. Since one of the key KDB requirements is version logging support (see Section 4), we also hold all the history of object states using the Django reversion⁷³ library. The Django application also handles authorization, as we also keep the editor usernames as a part of the stored history. The KDB service is the only service that has user-generated content; therefore this requires backing up the database in the long term. Since the data volume is quite low, of the order of just a few dozen thousand rows, we simply do a daily data dump through the Django interface and upload it to a Google Drive automatically. The source code is available on the GitHub.⁷⁴

7. Conclusions

This work describes the SNAD Viewer which aims to help address one of the most daunting challenges of contemporary science: how to make sense of big data sets generated by modern experiments. In many fields with strong observational ties, like astronomy, despite undeniable progress in automatic learning techniques, a significant fraction of potential breakthroughs are still expected to require human screening and intervention for the foreseeable future. We conceived the SNAD Viewer to optimize the allocation of human resources in such tasks.

Our initial goal was to centralize external information about the objects present in ZTF data releases through a single web interface, thus allowing SNAD experts to provide feedback to adaptive learning pipelines more efficiently. However, we evolved the Viewer to be a host of our team expertise, and now it also hosts a knowledge database containing thousands of annotations which we expect to provide valuable priors to inform the training and design of future learning algorithms.

Our system is currently based on a multi-service infrastructure, which allowed us to achieve good development speed, simplicity, and the required level of availability. We built it using the Dash framework, giving our users a powerful interactive interface. It is currently highlighted at “Plotly and Dash 500” rating,⁷⁵ and it is regularly used from all continents, but Antarctica, serving from a dozen to a few hundred unique visitors per day and responding to hundreds of thousands HTTP requests every month.

The SNAD Viewer framework has also proven to be resilient under different user conditions, and it is currently integrated into the ANTARES and Fink brokers, as well as into the Young Supernova Experiment marshal (Coulter et al.

⁶⁵ <https://github.com/snad-space/ztf-periodic-catalog-db>

⁶⁶ <https://github.com/snad-space/snad-tns>

⁶⁷ <https://github.com/light-curve/light-curve-feature>

⁶⁸ <https://rocket.rs>

⁶⁹ <https://github.com/snad-space/web-light-curve-features>

⁷⁰ <https://github.com/hombit/geo302>

⁷¹ <https://github.com/snad-space/ztf-fits-proxy>

⁷² <https://www.django-rest-framework.org>

⁷³ <https://pypi.org/project/django-reversion/>

⁷⁴ <https://github.com/snad-space/akb-backend>

⁷⁵ <https://dash-demo.plotly.host/plotly-dash-500/>

2022). Moreover, it has enabled the development of all the SNAD projects using ZTF data (e.g., Malanchev et al. 2021; Aleo et al. 2022; Pruzhinskaya et al. 2022), including the discovery of lost transient candidates in ZTF DR (see footnote 20).

In this era of big data, the SNAD Viewer is an illustrative example of the potential enabled by the open science policies of 21st century astronomy. Its development was only possible due to the large efforts allocated to the maintenance of public data archives and APIs, which guarantees accessibility of final data products to the entire astronomical community, and consequently optimizes the scientific results from these data. It is also a statement on the effort necessary to nurture a truly interdisciplinary environment. The development of its features was, from the very beginning, guided by domain experts who voiced their needs and concerns raised during the analysis process. This experience will certainly be valuable once data from LSST becomes available.

Finally, the SNAD Viewer framework described in this work was designed to allow easy adaptation as well as scalability to other surveys. In the era of LSST, such data centralization about specific objects will be as important to most astrophysical domains as it already is today for multi-messenger astronomy. However to realize this vision, tools used for different science cases must be designed to be easily adaptable to the requirements of different research sub-fields. With the SNAD Viewer we present an example of such a tool, and its successful experience so far also demonstrates the great potential they hold for the future of astronomical discovery.

Authors are grateful to Kirill Sokolovsky, Adam Scott, Julien Peloton, and Vadim Krushinsky for the helpful discussions. We thank all users of the Viewer for their feedback and bug reports. We thank Clara Heinrich for her work in the illustration of Figure 9.

This research has made use of the NASA/IPAC Infrared Science Archive, which is funded by the National Aeronautics and Space Administration and operated by the California Institute of Technology. This research has made use of the SIMBAD database, VizieR catalog, and “Aladin sky atlas” access tool, operated at CDS, Strasbourg, France. This work has made use of results from the ESA space mission Gaia, the data from which were processed by the Gaia Data Processing and Analysis Consortium (DPAC). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the Gaia Multilateral Agreement. Some of the authors are members of the Gaia Data Processing and Analysis Consortium (DPAC).

We used the equipment funded by the Lomonosov Moscow State University Program of Development. This work made use of server hosting services from the Donald Bren school of

Information and Computer Sciences of the University of California, Irvine, and of a machine acquired with the UCI 2021-2022 Professional Development Award. This work made use of the Illinois Campus Cluster, a computing resource that is operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA) and which is supported by funds from the University of Illinois at Urbana-Champaign.

The reported study was funded by RFBR according to the research project 20-02-00779. M.V.P. acknowledges the support by the Interdisciplinary Scientific and Educational School of Moscow University “Fundamental and Applied Space Research” E.E.O.I. and E.R. received financial support from CNRS International Emerging Actions under the project *Real-time analysis of astronomical data for the Legacy Survey of Space and Time* during 2021-2022. P.D.A. is supported by the Illinois Survey Science Graduate Fellowship from the Center for AstroPhysical Surveys (CAPS)⁷⁶ at the National Center for Supercomputing Applications (NCSA). A.K.M. acknowledges the support from the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through grants UID/FIS/00099/2019 for CENTRA and EXPL/FIS-AST/1368/2021. Supported by Nonprofit Foundation for the Development of Science and Education “Intellect.”

ORCID iDs

Konstantin Malanchev  <https://orcid.org/0000-0001-7179-7406>

Matwey V. Kornilov  <https://orcid.org/0000-0002-5193-9806>

Maria V. Pruzhinskaya  <https://orcid.org/0000-0001-7178-0823>

Emille E. O. Ishida  <https://orcid.org/0000-0002-0406-076X>

Patrick D. Aleo  <https://orcid.org/0000-0002-6298-1663>

Etienne Russeil  <https://orcid.org/0000-0001-9923-2407>

Sreevarsha Sreejith  <https://orcid.org/0000-0002-6423-1348>

Alberto Krone-Martins  <https://orcid.org/0000-0002-2308-6623>

References

- Abbott, T. M. C., Agüena, M., Alarcon, A., et al. 2022, *PhRvD*, **105**, 023520
- Aleo, P. D., Malanchev, K. L., Pruzhinskaya, M. V., et al. 2022, *NewA*, **96**, 101846
- Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., et al. 2022, *ApJ*, **935**, 167
- Bailer-Jones, C. A. L., Rybizki, J., Fouesneau, M., Demleitner, M., & Andrae, R. 2021, *AJ*, **161**, 147
- Bellm, E. C., Kulkarni, S. R., Barlow, T., et al. 2019a, *PASP*, **131**, 068003
- Bellm, E. C., Kulkarni, S. R., Graham, M. J., et al. 2019b, *PASP*, **131**, 018002
- Blagorodnova, N. 2017, *TNSCR*, **2017-1272**, 1
- Boch, T., & Fernique, P. 2014, in *ASP Conf. Ser. 485, Astronomical Data Analysis Software and Systems XXIII*, ed. N. Manset & P. Forshay (San Francisco, CA: ASP), 277

⁷⁶ <https://caps.ncsa.illinois.edu/>

- Bonnarel, F., Fernique, P., Bienaymé, O., et al. 2000, *A&AS*, **143**, 33
- Chen, X., Wang, S., Deng, L., et al. 2020, *ApJS*, **249**, 18
- Clementini, G., Ripepi, V., Molinaro, R., et al. 2019, *A&A*, **622**, A60
- Collette, A. 2013, Python and HDF5: Unlocking Scientific Data (Sebastopol, CA: O'Reilly Media, Inc.)
- Coulter, D. A., Jones, D. O., McGill, P., et al. 2022, YSE-PZ: An Open-source Target and Observation Management System, v0.3.0, Zenodo, doi:[10.5281/zenodo.7278430](https://doi.org/10.5281/zenodo.7278430)
- Dick, S. J. 2013, *Discovery and Classification in Astronomy: Controversy and Consensus* (Cambridge: Cambridge Univ. Press)
- Dowler, P., Bonnarel, F., Michel, L., & Demleitner, M. 2015, IVOA Data Link Version 1.0, IVOA Recommendation, doi:[10.5479/ADS/bib/2015ivoa.spec.0617D](https://doi.org/10.5479/ADS/bib/2015ivoa.spec.0617D)
- Duev, D. A., Mahabal, A., Masci, F. J., et al. 2019, *MNRAS*, **489**, 3582
- Eastman, J., Siverd, R., & Gaudi, B. S. 2010, *PASP*, **122**, 935
- Eyer, L., Audard, M., Holl, B., et al. 2022, arXiv:[2206.06416](https://arxiv.org/abs/2206.06416)
- Fielding, R., Gettys, J., Mogul, J., et al. 1999, RFC2616: Hypertext Transfer Protocol—HTTP/1.1, USA: RFC Editor
- Flewelling, H. A., Magnier, E. A., Chambers, K. C., et al. 2020, *ApJS*, **251**, 7
- Förster, F., Cabrera-Vives, G., Castillo-Navarrete, E., et al. 2021, *AJ*, **161**, 242
- Fremling, C., Miller, A. A., Sharma, Y., et al. 2020, *ApJ*, **895**, 32
- Gaia Collaboration, Prusti, T., de Bruijne, J. H. J., et al. 2016, *A&A*, **595**, A1
- Gaia Collaboration, Vallenari, A., Brown, A. G. A., et al. 2022, arXiv:[2208.00211](https://arxiv.org/abs/2208.00211)
- Ginsburg, A., Sipőcz, B. M., Brasseur, C. E., et al. 2019, *AJ*, **157**, 98
- Gonzalez-Núñez, J., Salgado, J., Gutiérrez-Sánchez, R., et al. 2019, in ASP Conf. Ser. 523, *Astronomical Data Analysis Software and Systems XXVII*, ed. P. J. Teuben et al. (San Francisco, CA: ASP), 417
- Green, G. 2018, *JOSS*, **3**, 695
- Green, G. M., Schlafly, E., Zucker, C., Speagle, J. S., & Finkbeiner, D. 2019, *ApJ*, **887**, 93
- Green, G. M., Schlafly, E. F., Finkbeiner, D. P., et al. 2015, *ApJ*, **810**, 25
- Guilloc'hon, J., Parrent, J., Kelley, L. Z., & Margutti, R. 2017, *ApJ*, **835**, 64
- Hambleton, K. M., Bianco, F. B., Street, R., et al. 2022, arXiv:[2208.04499](https://arxiv.org/abs/2208.04499)
- Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, *Natur*, **585**, 357
- Heinze, A. N., Tonry, J. L., Denneau, L., et al. 2018, *AJ*, **156**, 241
- Hunter, J. D. 2007, *CSE*, **9**, 90
- Ishida, E. E. O., Komilov, M. V., Malanchev, K. L., et al. 2021, *A&A*, **650**, A195
- Jayasinghe, T., Stanek, K. Z., Kochanek, C. S., et al. 2019, *MNRAS*, **485**, 961
- Jones, D. O., Foley, R. J., Narayan, G., et al. 2021, *ApJ*, **908**, 143
- Jones, D. O., Scolnic, D. M., Riess, A. G., et al. 2018, *ApJ*, **857**, 51
- Kochanek, C. S., Shappee, B. J., Stanek, K. Z., et al. 2017, *PASP*, **129**, 104502
- Kosakowski, A., Kilic, M., Brown, W. R., Bergeron, P., & Kupfer, T. 2022, *MNRAS*, **516**, 720
- Kuhn, M. A., de Souza, R. S., Krone-Martins, A., et al. 2021, *ApJS*, **254**, 33
- Kupfer, T., Prince, T. A., van Roestel, J., et al. 2021, *MNRAS*, **505**, 1254
- Lomb, N. R. 1976, *Ap&SS*, **39**, 447
- LSST Science Collaboration, Abell, P. A., Allison, J., et al. 2009, arXiv:[0912.0201](https://arxiv.org/abs/0912.0201)
- Lyke, B. W., Higley, A. N., McLane, J. N., et al. 2020, *ApJS*, **250**, 8
- Malanchev, K. 2021, Light-curve: Light Curve Analysis Toolbox, Astrophysics Source Code Library, ascl:[2107.001](https://ascl.net/2107.001)
- Malanchev, K. L., Pruzhinskaya, M. V., Korolev, V. S., et al. 2021, *MNRAS*, **502**, 5147
- Mandel, E., & Vikhlinin, A. 2022, ericmandel/js9, v3.6.2, Zenodo, doi:[10.5281/zenodo.596052](https://doi.org/10.5281/zenodo.596052)
- Matheson, T., Stubens, C., Wolf, N., et al. 2021, *AJ*, **161**, 107
- McKinney, W. 2010, in Proc. 9th Python in Science Conf., ed. S. van der Walt & J. Millman, 56
- Möller, A., Peloton, J., Ishida, E. E. O., et al. 2021, *MNRAS*, **501**, 3272
- Nordin, J., Brinell, V., van Santen, J., et al. 2019, *A&A*, **631**, A147
- Ochsenbein, F., Bauer, P., & Marcout, J. 2000, *A&AS*, **143**, 23
- 2020, The pandas development team/pandas-dev/pandas: Pandas, latest, v1.5.3, Zenodo, doi:[10.5281/zenodo.3509134](https://doi.org/10.5281/zenodo.3509134)
- Perley, D. A., Fremling, C., Sollerman, J., et al. 2020, *ApJ*, **904**, 35
- Pruzhinskaya, M. V., Ishida, E. E. O., Novinskaya, A. K., et al. 2022, arXiv:[2208.09053](https://arxiv.org/abs/2208.09053)
- Pruzhinskaya, M. V., Malanchev, K. L., Kornilov, M. V., et al. 2019, *MNRAS*, **489**, 3591
- Riello, M., De Angeli, F., Evans, D. W., et al. 2021, *A&A*, **649**, A3
- Rodriguez, A. C., Mróz, P., Kulkarni, S. R., et al. 2022, *ApJ*, **927**, 150
- Sahr, K., White, D., & Kimerling, A. J. 2003, *Cartography and Geographic Information Science*, **30**, 121
- Samus', N. N., Goranskii, V. P., Durlevich, O. V., et al. 2003, *AstL*, **29**, 468
- Samus', N. N., Kazarovets, E. V., Durlevich, O. V., Kireeva, N. N., & Pastukhova, E. N. 2017, *ARep*, **61**, 80
- Sánchez-Sález, P., Lira, H., Martí, L., et al. 2021, *AJ*, **162**, 206
- Scargle, J. D. 1982, *ApJ*, **263**, 835
- Schlafly, E. F., & Finkbeiner, D. P. 2011, *ApJ*, **737**, 103
- Schlegel, D. J., Finkbeiner, D. P., & Davis, M. 1998, *ApJ*, **500**, 525
- Stern, D., Djorgovski, S. G., Krone-Martins, A., et al. 2021, *ApJ*, **921**, 42
- Szalay, A., & Gray, J. 2001, *Sci*, **293**, 2037
- Udalski, A. 2003, *AcA*, **53**, 291
- Udalski, A., Szymański, M. K., & Szymański, G. 2015, *AcA*, **65**, 1
- Vincenzi, M., Sullivan, M., Möller, A., et al. 2022, *MNRAS*, **518**, 1106
- Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, *NatMe*, **17**, 261
- Watson, C. L., Henden, A. A., & Price, A. 2006, *SASS*, **25**, 47
- Wenger, M., Ochsenbein, F., Egret, D., et al. 2000, *A&AS*, **143**, 9
- Zhou, R., Newman, J. A., Mao, Y.-Y., et al. 2021, *MNRAS*, **501**, 3309
- Zouca, A., Singer, L., Lenz, D., et al. 2019, *JOSS*, **4**, 1298