LSST:UK Consortium

LSST:UK Science Centre

D3.2.2 Lasair User Interface

WP3.2 LASAIR–the UK transient broker for LSST

Project Acronym	LUSC-B
Project Title	UK Involvement in the Legacy Survey of Space and Time
Document Number	LUSC-B-32

Submission date	30/AUG/22
Version	1.0.
Status	Final
Author(s) inc. institutional affiliation	Roy Williams and Gareth Francis, University of Edinburgh Ken Smith and Dave Young, Queens University Belfast
Reviewer(s)	S. Caswell (Leicester, C. Inserra(Cardiff)
Dissemination level	Public

Version History

Version	Date	Comments, Changes, Status	Authors, contributors, reviewers
0.1	16/FEB/22	First draft from Williams	R. Williams
0.2	30/AUG/22	Contains updates from Young, Francis and Smith	D. Young
0.3	06/SEP/22	Accepted track changes, updated document properties	T Sloan
0.4	15/Oct/22	Modified from reviews	R Williams, D Young
1.0	18/NOV/22	Final following Exec Group approval	T. M .Sloan

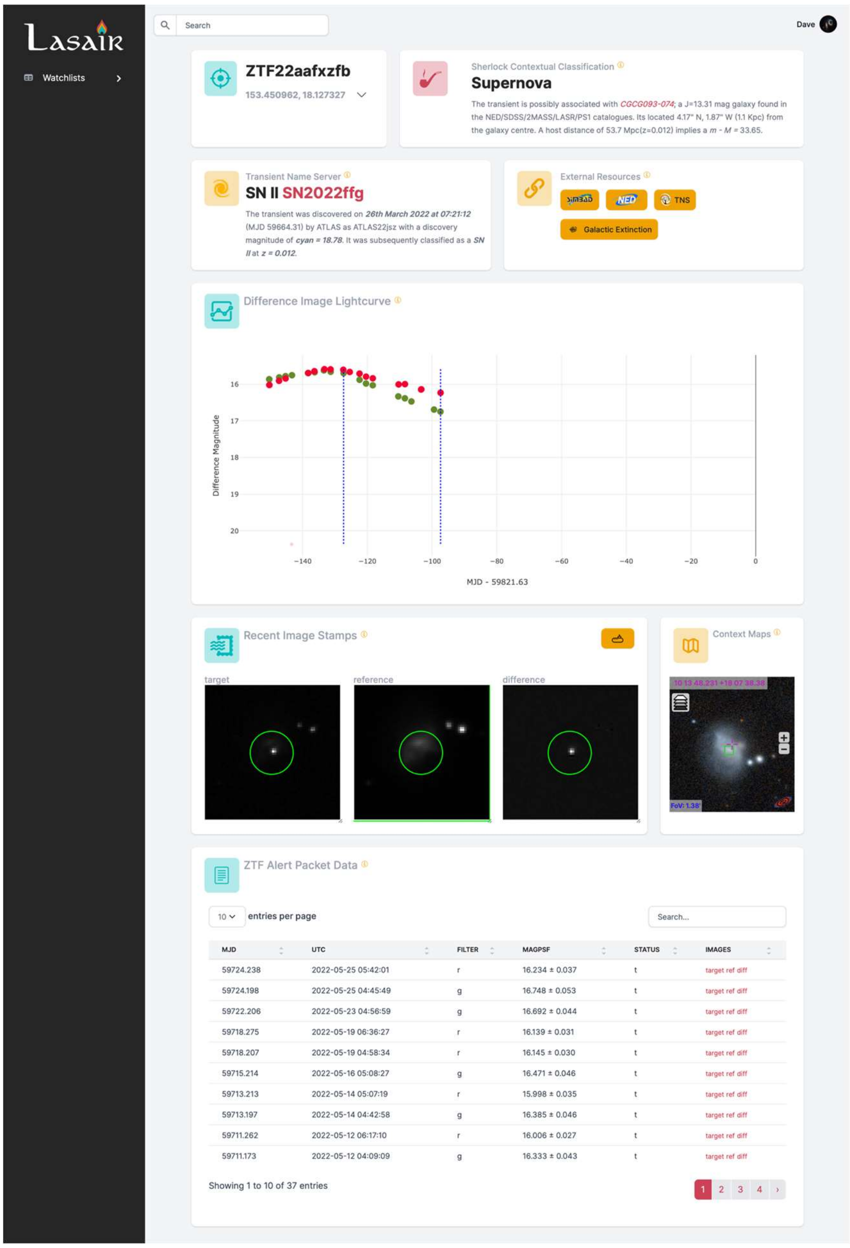
Table of Contents

1 Executive Summary

This report describes the user interface of the Lasair Community Broker: the purpose, the architecture and design, the underlying technologies, and how the components work together.

2 What is Lasair?

The Lasair Community Broker[1] is a platform for astronomers to work effectively with the LSST transient alert stream: it is designed to be fast, flexible, and capable. Given that LSST data is not yet available, Lasair has worked with data from the Zwicky Transient Facility (ZTF)[2] for the last four years. ZTF is similar in crucial ways to what the LSST stream[3] will be: data is delivered as fast as possible from the telescope, using the Kafka technology. We have benefitted enormously from prototyping with ZTF, and



The transient is possibly associated with *CGCG093-074*; a J=13.31 mag galaxy found in the NED/SDSS/2MASS/LASR/PS1 catalogues. Its located 4.17" N, 1.87" W (1.1 Kpc) from the galaxy centre. A host distance of 53.7 Mpc(z=0.012) implies a *m* - *M* = 33.65.

The transient was discovered on ***26th March 2022 at 07:21:12*** (MJD 59664.31) by ATLAS as ATLAS22jsz with a discovery magnitude of ***cyan = 18.78***. It was subsequently classified as a ***SN II*** at ***z = 0.012***.

MJD	UTC	FILTER	MAGPSF	STATUS	IMAGES
59724.238	2022-05-25 05:42:01	r	16.234 ± 0.037	t	target ref diff
59724.198	2022-05-25 04:45:49	g	16.748 ± 0.053	t	target ref diff
59722.206	2022-05-23 04:56:59	g	16.692 ± 0.044	t	target ref diff
59718.275	2022-05-19 06:36:27	r	16.139 ± 0.031	t	target ref diff
59718.207	2022-05-19 04:58:34	r	16.145 ± 0.030	t	target ref diff
59715.214	2022-05-16 05:08:27	g	16.471 ± 0.046	t	target ref diff
59713.213	2022-05-14 05:07:19	r	15.998 ± 0.035	t	target ref diff
59713.197	2022-05-14 04:42:58	g	16.385 ± 0.046	t	target ref diff
59711.262	2022-05-12 06:17:10	r	16.006 ± 0.027	t	target ref diff
59711.173	2022-05-12 04:09:09	g	16.333 ± 0.043	t	target ref diff

Showing 1 to 10 of 37 entries

Figure 1. The Lasair Object page provides the three cutout images of the object, lightcurves in all filters, AladinLite panchromatic views, the Sherlock context, and relevant information from the Transient Name Service (TNS).

we are sure that the infrastructure we have built puts us in a good position for handling the LSST stream.

The central purposes of Lasair are to (1) collect the transient alerts and amplify their scientific usefulness by adding value, and (2) to allow scientists to find objects that are relevant to their own science goals. Figure 1 shows the Lasair object display page; each alert contains a fresh point of the light curve ("candidate" in ZTF, and "diaSource" in LSST), and the Lasair database shows (astrophysical) objects as collections of detections, together with the latest cutout images. Lasair adds contextual information from the "Sherlock" transient classification algorithm (see below), and matches with the TNS – the IAU record of detected supernovae [4].

While other LSST community brokers are based on existing classification algorithms, Lasair is not. Rather Lasair is a platform to annotate alerts in many ways, filter the alerts, and deliver the results to astronomers. Instead of a classifier, we have made the 'annotator' system, so that a user can run their own classifier on Lasair lightcurves and push the result back into the system. We have one such classifier now in operation, called FastFinder.

2.1 Future Changes

Much of the Lasair infrastructure and user interface is built, based on the prototype alert stream from ZTF. However, our real purpose is to be a broker for LSST alerts, which are different in several ways. The most obvious difference is in data bulk – perhaps 30 time more – but we expect the Lasair user interface to be different in a few ways:

- LSST brightness will be written as flux (nanoJansky) instead of magnitudes.
- Six light filters instead of two, thus changing the lightcurve features.
- In the first year at least, the unprecedented depth of LSST means no comprehensive galaxy catalogues will be available.
- LSST alerts will include an object packet (diaObject) with LSST-provided lightcurve features.
- LSST alerts will include forced photometry packets (diaForcedSources).
- Some LSST alerts will include a solar system packet (ssObject).
- Some data will be proprietary, available only to data-rights holders.
- Some alert data will not come along with the alert, but only in subsequent days.
- A new scheme so that users can request forced photometry.

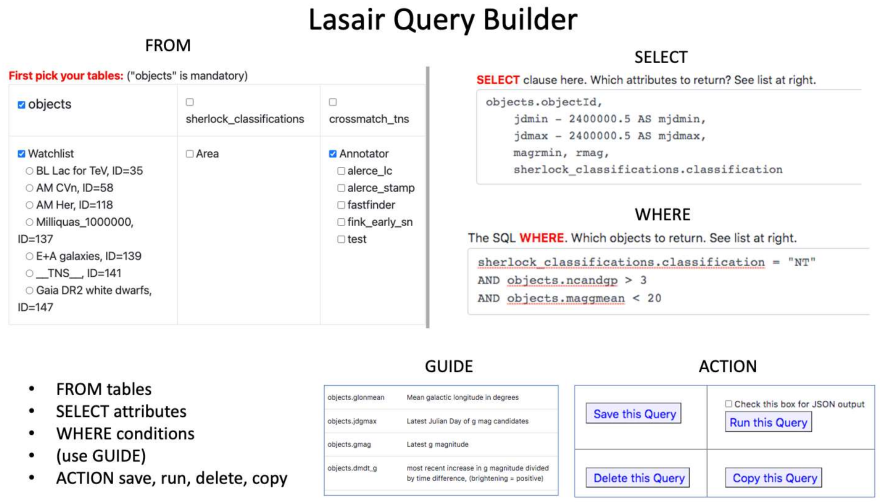
All these qualitative changes will mean changes to the Lasair user interface.

3 SQL queries

For finding "interesting" objects, Lasair offers a SQL-based query system, that includes several auxiliary tables that can be joined. Users can select on several additional resources in addition to the features of the astrophysical object; including sky position, proposed host galaxy, TNS metadata.

Lasair also has an "annotation" capability, so that external entities can contribute data, that can in turn be used for selections. When users build filters, watchlists, areas, and annotations, they can be private; or a user can allow anyone else to share by seeing and utilising that resource.

Figure 2 sketches the filter builder. The FROM section is a menu of tables from which the filter will be built, including the Sherlock information, watchlists and areas, the TNS database, and the available annotator components. The SELECT clause is an SQL phrase expressing what will be returned, and WHERE phrase is a set of cuts to get only what is interesting. There is a guide showing all available attributes, and a set of actions such as running the filter, saving it, or copying a filter from another user – for modification.



Lasair Query Builder

FROM

First pick your tables: ("objects" is mandatory)

☑ objects	□ sherlock_classifications	□ crossmatch_tns
☑ Watchlist ○ BL Lac for TeV, ID=35 ○ AM CVn, ID=58 ○ AM Her, ID=118 ○ Milliquas_1000000, ID=137 ○ E+A galaxies, ID=139 ○ __TNS__, ID=141 ○ Gaia DR2 white dwarfs, ID=147	□ Area	☑ Annotator □ alerce_lc □ alerce_stamp □ fastfinder □ fink_early_sn □ test

SELECT

SELECT clause here. Which attributes to return? See list at right.

```
objects.objectId,
    jdmin - 2400000.5 AS mjdmin,
    jdmax - 2400000.5 AS mjdmax,
    magrmin, rmag,
    sherlock_classifications.classification
```

WHERE

The SQL WHERE. Which objects to return. See list at right.

```
sherlock_classifications.classification = "NT"
AND objects.ncandgp > 3
AND objects.maggmean < 20
```

- FROM tables
- SELECT attributes
- WHERE conditions
- (use GUIDE)
- ACTION save, run, delete, copy

GUIDE

objects.glonmean	Mean galactic longitude in degrees
objects.jdgmax	Latest Julian Day of g mag candidates
objects.gmag	Latest g magnitude
objects.dmdt_g	most recent increase in g magnitude divided by time difference, (brightening = positive)

ACTION

Figure 2. The Lasair query builder web page. The user creates a query by selecting source tables (objects, sherlock, watchlists, areas, TNS and annotations) then builds SQL-like SELECT and WHERE clauses. Queries can be saved, copied, and modified. A query can run real time as a filter, which produces an output stream.

A filter query can be typed into the web form and simply executed. After checking the syntax, a time limit and row limit is added, and the filter can be executed. If a user is logged in to Lasair, they can save the filter for future use; and they can also check a box for it to be run in real-time on the incoming alerts. Active filters pass only the alerts that interest that user; the output of a filter is a *stream*, which can be delivered to the user through a public Kafka system, or by email.

4 Watchlists and Watchmaps

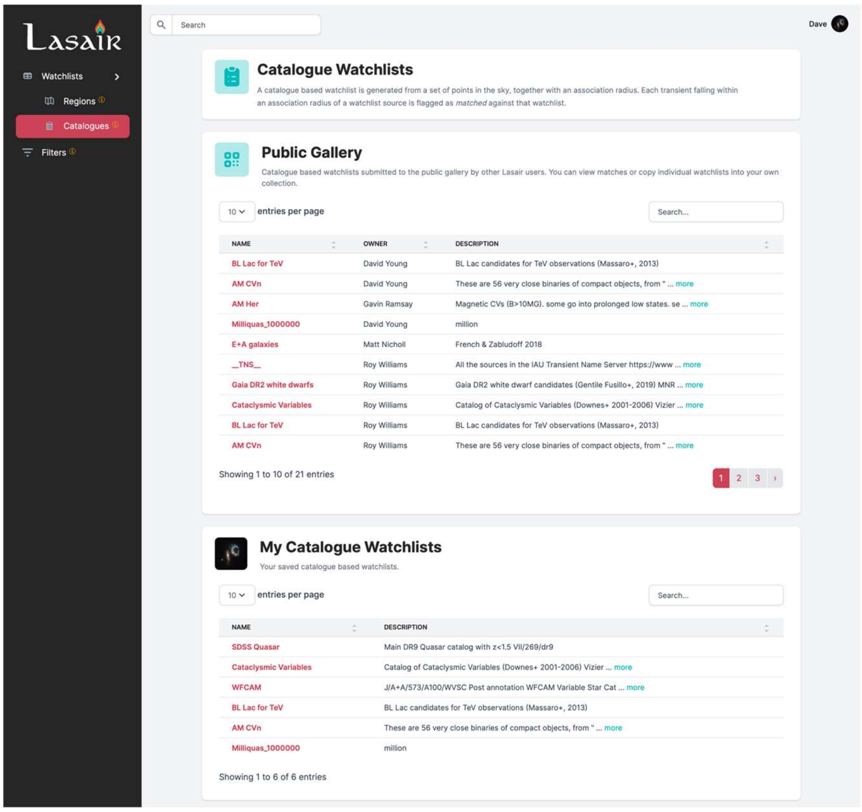
Lasair also offers discrimination based on the location of an alert in the sky. A watchlist is set of named sources of interest to a user, a "personal catalogue". Each source comes with a tolerance radius for the crossmatch, and is thus called a cone. When an alert falls in one of the cones, a tag record is made connecting the alert and the name of the cone. In this way, a user can declare interest in variability from their personal catalogue of sources, for example the million-quasar catalogue, or the 55 very close binaries of AM CVn type. Watchlists can contain millions of sources.

Users can also select alerts from a specific area of the sky, that we call a "Watchmap", for example a gravitational wave footprint, or the SDSS Stripe 82. Alerts that fall in the area are tagged, and a query/filter that selects on the watchmap tags will get just those. Both the watchlist and the watchmap capability are implemented with the MOC system [5]. For the area, the user uploads the MOC file itself, and for the watchlist, these files are built from the list of cones – see "Background Services" below.

5 Sherlock

Elucidating the nature of extra-galactic transients depends critically on identifying the host galaxy. Lasair achieves this by integration with a crossmatch system that has all the latest and deepest catalogues.

Sherlock [6] is a software package and integrated massive database system that provides a rapid and reliable spatial cross-match service for any astrophysical variable or transient. It associates the position of a transient with locally curated major astronomical catalogues (NED, PanStarrs, Gaia, 2MASS, SDSS, etc) and assigns a basic classification to the transient based on a boosted decision tree algorithm. At its most basic, it separates stars, AGN and supernova-like transients. The algorithm is queried via an API, which facilitates delegation of result storage to the calling code to do with it whatever is required (e.g. store the basic summary result but not the associated ranked catalogue entries, or store everything). Each Sherlock node is a multi-terabyte read-only replica, so addition of more context classification capacity to Lasair is just a matter of increasing the number of Sherlock nodes.



Catalogue Watchlists

A catalogue based watchlist is generated from a set of points in the sky, together with an association radius. Each transient falling within an association radius of a watchlist source is flagged as *matched* against that watchlist.

Public Gallery

Catalogue based watchlists submitted to the public gallery by other Lasair users. You can view matches or copy individual watchlists into your own collection.

10 entries per page — Search...

NAME	OWNER	DESCRIPTION
BL Lac for TeV	David Young	BL Lac candidates for TeV observations (Massaro+, 2013)
AM CVn	David Young	These are 56 very close binaries of compact objects, from " ... more
AM Her	Gavin Ramsay	Magnetic CVs (B>10MG). some go into prolonged low states. se ... more
Milliquas_1000000	David Young	million
E+A galaxies	Matt Nicholl	French & Zabludoff 2018
__TNS__	Roy Williams	All the sources in the IAU Transient Name Server https://www ... more
Gaia DR2 white dwarfs	Roy Williams	Gaia DR2 white dwarf candidates (Gentile Fusillo+, 2019) MNR ... more
Cataclysmic Variables	Roy Williams	Catalog of Cataclysmic Variables (Downes+ 2001-2006) Vizier ... more
BL Lac for TeV	Roy Williams	BL Lac candidates for TeV observations (Massaro+, 2013)
AM CVn	Roy Williams	These are 56 very close binaries of compact objects, from " ... more

Showing 1 to 10 of 21 entries — 1 2 3 ›

My Catalogue Watchlists

Your saved catalogue based watchlists.

10 entries per page — Search...

NAME	DESCRIPTION
SDSS Quasar	Main DR9 Quasar catalog with z<1.5 VII/269/dr9
Cataclysmic Variables	Catalog of Cataclysmic Variables (Downes+ 2001-2006) Vizier ... more
WFCAM	J/A+A/573/A100/WVSC Post annotation WFCAM Variable Star Cat ... more
BL Lac for TeV	BL Lac candidates for TeV observations (Massaro+, 2013)
AM CVn	These are 56 very close binaries of compact objects, from " ... more
Milliquas_1000000	million

Showing 1 to 6 of 6 entries

Figure 3. List of watchlists (public and user owned). A catalogue based watchlist is generated from a set of points in the sky, together with an association radius. Each transient falling within an association radius of a watchlist source is flagged as *matched* against that watchlist. If a watchlist is "active", it is crossmatched with incoming alerts in real time; if it is "public" then Lasair other users can see it in the public gallery.

6 User Interface

Lasair allows users to execute a wide class of SQL queries on the database of active objects, and to convert those to filters on the real-time stream, with the outputs pushed – in near real time – to other machines or to email. Lasair also provides an API for data access, so that users can build code and notebooks to work with Lasair objects and lightcurves. Lasair allows users to "annotate" the database of transients with their own classifications. Lasair allows users to log in to the website, where they can store and share filters, watchlists, areas, and annotations.

6.1 Web

The Lasair webserver and API server allow users to control their interactions with the alert database and the stream. Information about an astrophysical object can be displayed on a web page; users can create, run save, and share filters; users can create and share watchlists and areas; and users can create annotators to classify light curves or other information and push the classifications to the Lasair database, where they are available for querying.

The web server is built with the Django software [7]: requests are passed to python code, data added from the databases, and the results sent to templates that make web pages. The web server is built in two layers: a low-security "http" system that is easy to manage, but accessible only to the internal network; then an outward-facing secure proxy uses "https" and certificates and forwards requests to the internal system.

There is also a Lasair API, where machines make requests and receive responses; the API has a token-based throttling system, so that any individual user cannot overwhelm the system with requests.

As with any Django web server, there is also has an "admin" page accessible to administrators, allowing changes to user's accounts, seeing all the stored resources (private or public), and administering annotators to prevent abuse.

The Lasair API has the following methods:

- cone: runs a cone search on all the objects in the Lasair database.
- query: runs a SQL SELECT query on the Lasair database.
- streams: returns a record of the output from a Lasair streaming query.
- objects: returns a machine-readable version of the object web page.
- lightcurves: returns simple lightcurves for a number of objects.
- sherlock/objects: returns Sherlock information about a list of named objects.
- sherlock/position: returns Sherlock information about a sky position.

6.2 Annotator

The Lasair API supports *annotation*: a structured external packet of extra information about a given object, that is stored in the annotations table in the SQL database. We have worked in collaboration with the teams of other LSST community brokers -- Fink and Alerce -- to ingest their classifications that are distributed via Kafka (see "Background Services" below). Thus a Lasair user can, for example, create a stream of alerts where (a) Sherlock sees an associated galaxy and (b) Fink sees a probability > 0.5 that the alert is an early supernova. The "Annotations" checkbox can be seen in the filter builder as shown in Figure 2.

Annotations can also be generated from a Lasair stream. The FastFinder annotator uses a "pre-query" within Lasair to generate a stream, and for each one the lightcurve is read in via the API, and classified, with the results being pushed back to Lasair as above.

Since annotation puts data into the Lasair database, we must be careful of who is allowed to do this. We distinguish between *annotator* and *annotation*: the former identifies who is responsible and the description, while the latter references from which annotator it is derived. Whenever an annotation is received by the Lasair API, it should contain the unique API token of the owner of that annotator, and if there is a mismatch, it will be rejected. Annotators are managed through the admin page described above.

6.3 SQL database

The SQL database drives the webserver and API through Django, and also executes user queries on the object table, joined with other tables. There are some precautions and modifications applied to the SELECT and WHERE clauses that the user provides:

- User-written queries (filters) run on a read-only account on the database
- An execution time limit is added
- A limit on output rows is added
- Some words are prohibited: some examples are `create, select, from, where, join, inner, outer, with, union, exists.`
- There is a careful parsing of parentheses and brackets.

The important tables of the SQL database are shown in Figure 4. Tables connected by blue lines are user-owned resources – annotators, filters, watchlists, areas. Tables connected by red lines are about astrophysical objects and can be used in a user-built filters. The very large 'candidates' table is not in the SQL database, but rather in the NoSQL database – Cassandra is used as a lightcurve store.

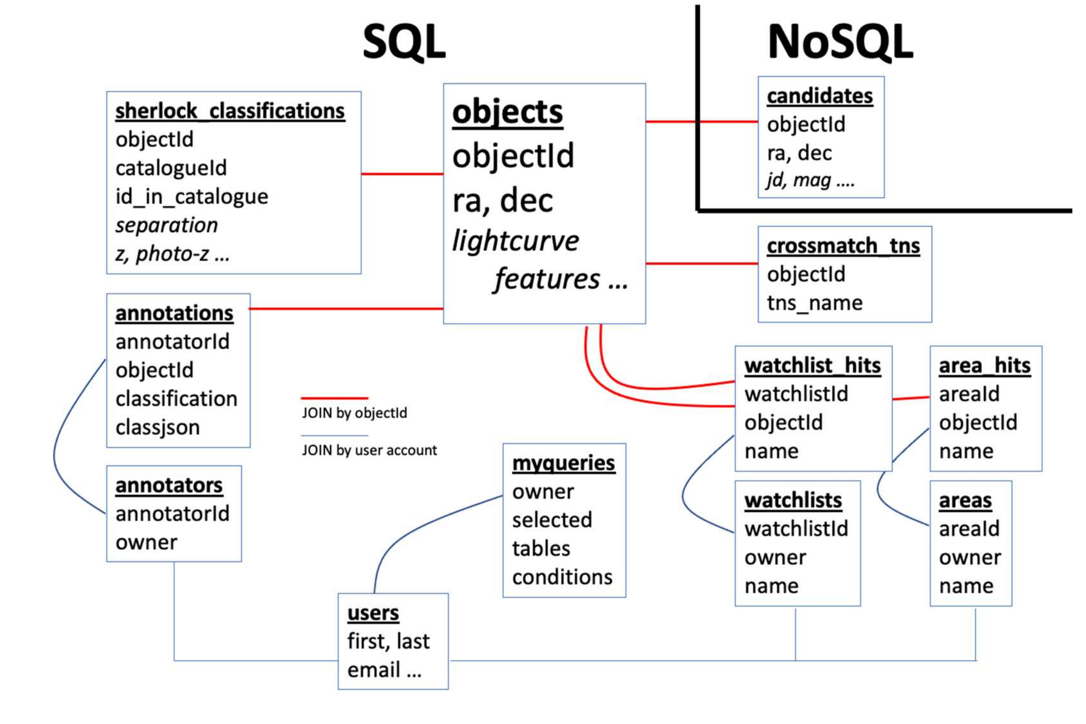


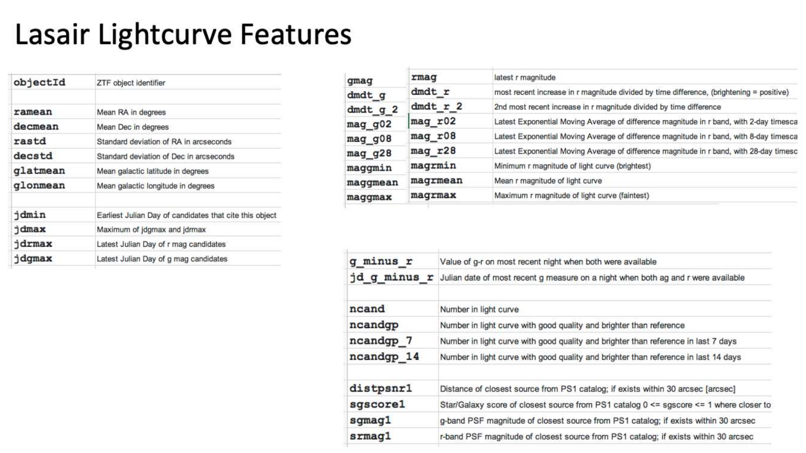
Figure 4. The Lasair entity relationship diagram. Candidates are stored in a separate NoSQL system from the other tables that are in an SQL system. Queries are built with the "objects" table joined with others. The candidates table is not directly accessible with SQL queries. Sherlock, TNS, and annotation tables provide extra information. Users can create stored queries, watchlists, areas, and annotators keyed to their user account.

The attributes of the object table include a number of "features" of the lightcurve, that users can use to select interesting objects. These are not computed on the whole lightcurve (~4 years), but only on the most recent 30 days, the candidates included with each alert: this is so that alerts can be handled in parallel, without need for multiple filter nodes to be continually hitting the databases.

The feature list as of July 2022 is shown in Figure 5. Spacetime is in the left panel: the distribution of the measured positions for each candidate, galactic coordinates, and the times of the start and end of the lightcurve. Other features include the distribution of magnitudes, moving averages of the magnitudes, numbers of candidates, and information about nearest PanStarrs objects.

Much of the Lasair user interface is fairly solid – the ideas of building filters with lightcurve features, the watchlists and watchmaos, etc – one part that is still under development is the nature of these features. The ultimate objective of Lasair is to be a community broker for LSST, and the ZTF implementation described in this report is simply a prototype. At this time (2022) we are looking to the lightcurve features relevant for LSST. In contrast to ZTF, LSST will provide a large set of features [8] to describe periodic and stochastic variability. However, there is not much on recent, rapid transients.

Our current thinking is to characterise so-called "tiny lightcurves", which have only a few points from a few nights, presumably from an explosive transient like SN, KN, TDE, or GRB. For a single-filter LC, we have tried a number of fitting methods: linear, quadratic, t*exp(-t), Bazin, etc, as well as gaussian process. Each has their advantages. But we are also experimenting with features that consider all the filters together, given that the 6 filters of LSST is a lot more than the 2 filters of ZTF. We are trying to estimate three things: (1) how fast is the source brightening, (2) color evolution, and (3) absolute magnitude.



Lasair Lightcurve Features

objectId	ZTF object identifier
ramean	Mean RA in degrees
decmean	Mean Dec in degrees
rastd	Standard deviation of RA in arcseconds
decstd	Standard deviation of Dec in arcseconds
glatmean	Mean galactic latitude in degrees
glonmean	Mean galactic longitude in degrees
jdmin	Earliest Julian Day of candidates that cite this object
jdmax	Maximum of jdgmax and jdrmax
jdrmax	Latest Julian Day of r mag candidates
jdgmax	Latest Julian Day of g mag candidates

gmag	rmag	latest r magnitude
dmdt_g	dmdt_r	most recent increase in r magnitude divided by time difference, (brightening = positive)
dmdt_g_2	dmdt_r_2	2nd most recent increase in r magnitude divided by time difference
mag_g02	mag_r02	Latest Exponential Moving Average of difference magnitude in r band, with 2-day timesca
mag_g08	mag_r08	Latest Exponential Moving Average of difference magnitude in r band, with 8-day timesca
mag_g28	mag_r28	Latest Exponential Moving Average of difference magnitude in r band, with 28-day timesc
maggmin	magrmin	Minimum r magnitude of light curve (brightest)
maggmean	magrmean	Mean r magnitude of light curve
maggmax	magrmax	Maximum r magnitude of light curve (faintest)

g_minus_r	Value of g-r on most recent night when both were available
jd_g_minus_r	Julian date of most recent g measure on a night when both ag and r were available
ncand	Number in light curve
ncandgp	Number in light curve with good quality and brighter than reference
ncandgp_7	Number in light curve with good quality and brighter than reference in last 7 days
ncandgp_14	Number in light curve with good quality and brighter than reference in last 14 days
distpsnr1	Distance of closest source from PS1 catalog; if exists within 30 arcsec [arcsec]
sgscore1	Star/Galaxy score of closest source from PS1 catalog 0 <= sgscore <= 1 where closer to
sgmag1	g-band PSF magnitude of closest source from PS1 catalog; if exists within 30 arcsec
srmag1	r-band PSF magnitude of closest source from PS1 catalog; if exists within 30 arcsec

Figure 5. Lasair lightcurve features as of July 2022.

7 Acronyms

API: Application Programming Interface

IAU: International Astronomical Union

LSST: Legacy Survey of Space and Time

MOC: Multi-Order Coverage

SQL: Structured Query Language

TNS: Transient Name Service

ZTF: Zwicky Transient Facility

8 Glossary of Terms

Term	Definition
Alert Broker	A platform and software system designed to read transient alerts from the Rubin Observatory. A broker will redistribute these alerts to other platforms, brokers and to the broader scientific community. Before redistribution, a broker may inject value-added content into the alert packets.
Annotate	To add value-added content to an transient alert packet.
Crossmatch (or conesearch)	The sky coordinates of an astrophysical object or objects are matched (or cross-matched) against another list of sky coordinates (e.g. a galaxy catalogue). Sources matched within a given separation tolerance (search radius) are returned as an associated pair.
Filter	Based on a SQL query, a user can write a filter to strictly define the properties of transients they are particularly interested in. Transients matching these criteria will be found by the filter. Filters can be saved and rerun.
Kafka	A distributed streaming platform used to build real-time streaming data pipelines. Rubin's main alert stream pipeline and Lasair both rely on Kafka to send (and receive in the case of Lasair) alert packets.
Lasair	Pronounced '*L-AH-s-uh-r*'. Means flame or flash in Scots and Irish Gaelic.
Sherlock	A large database of historical astronomical catalogues combined with an association algorithm used to predict the nature of a transient event based on its local context.
Stream	Real-time distribution of transient matches (whether via filters, watchlists or watchmaps) typically via the Kafka streaming platform.
Watchlist	A set/catalogue of sky positions and associated radii. Transients found within an association radius of a watchlist sky-position are 'matched' against the watchlist source. Users can save watchlists and be alerted to new matches as new transients are detected.
Watchmap	A user-defined region of the sky. Transients discovered within the region are 'matched' against the watchmap. Users can save watchmaps and be alerted to new matches as new transients are detected.

9 References

[1] Lasair Community Broker: https://lasair-ztf.lsst.ac.uk

[2] Zwicky Transient Facility: https://ztf.caltech.edu/

[3] LSST alert stream: https://www.lsst.org/scientists/alert-brokers

[4] Transient Naming System: https://www.wis-tns.org/

[5] Multi-Order Coverage Maps:
https://www.ivoa.net/documents/MOC/ and https://pypi.org/project/mocpy/

[6] Sherlock system: https://arxiv.org/abs/2003.09052

[7] Django web framework: https://www.djangoproject.com/

[8] E. Bellm, Review of Timeseries Features: https://dmtn-118.lsst.io/