Babamul & BOOM

Next Generation alert brokering in the ZTF + Rubin era

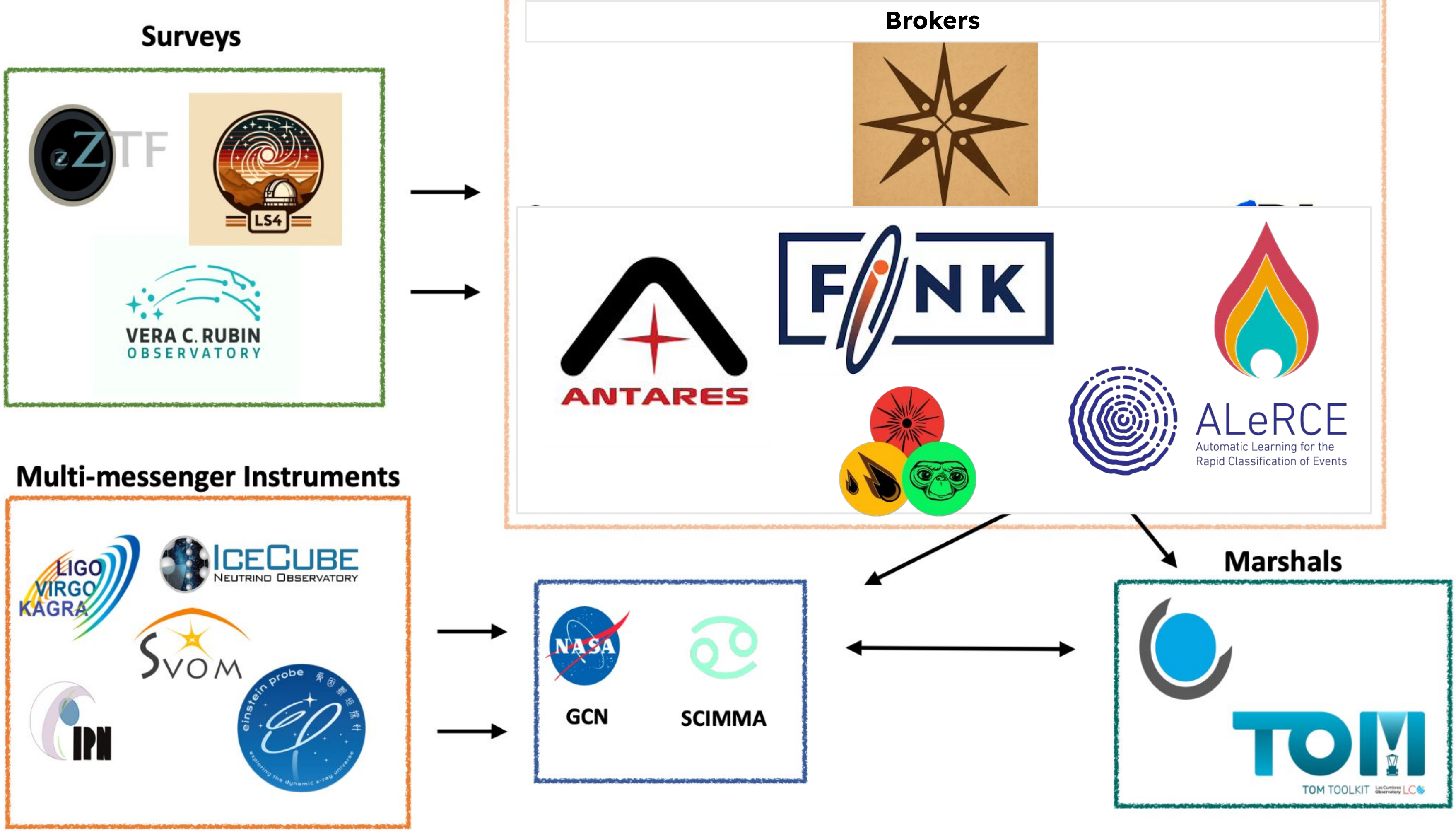
Michael Coughlin (University of Minnesota)

Team

Theophile Du Laz
Pete Bachant
Thomas Culino
Matthew Graham
Mansi Kasliwal

Michael Coughlin
Jake Simones
Matthew Jenson
Alexandra Junell
Argyro Sasli
Felipe Nunes
Maojie Xu
Benny Border
Antoine Le Calloch
Sushant Sharma Chaudhary

Josh Bloom
Stefan Van Der Walt



Surveys

Brokers

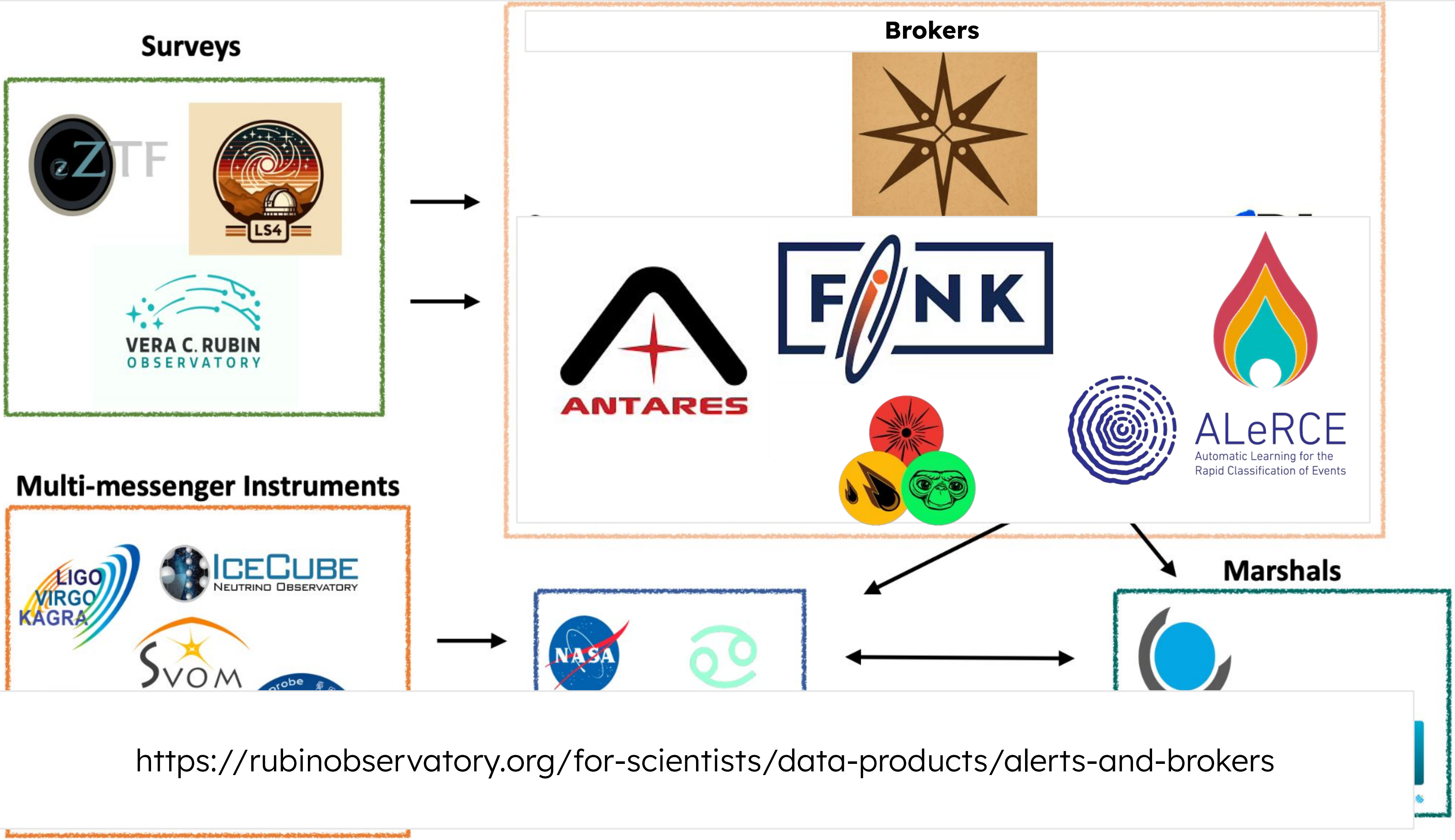
Multi-messenger Instruments

GCN

SCIMMA

Marshals

ZTF
LS4
VERA C. RUBIN OBSERVATORY
ANTARES
FINK
ALeRCE
Automatic Learning for the Rapid Classification of Events
LIGO VIRGO KAGRA
ICECUBE NEUTRINO OBSERVATORY
SVOM
IPN
einstein probe
exploring the dynamic x-ray universe
NASA
TOM
TOM TOOLKIT


ZTF
LS4
VERA C. RUBIN OBSERVATORY
ANTARES
FINK
ALeRCE
Automatic Learning for the Rapid Classification of Events
LIGO VIRGO KAGRA
ICECUBE NEUTRINO OBSERVATORY
SVOM
NASA

Surveys

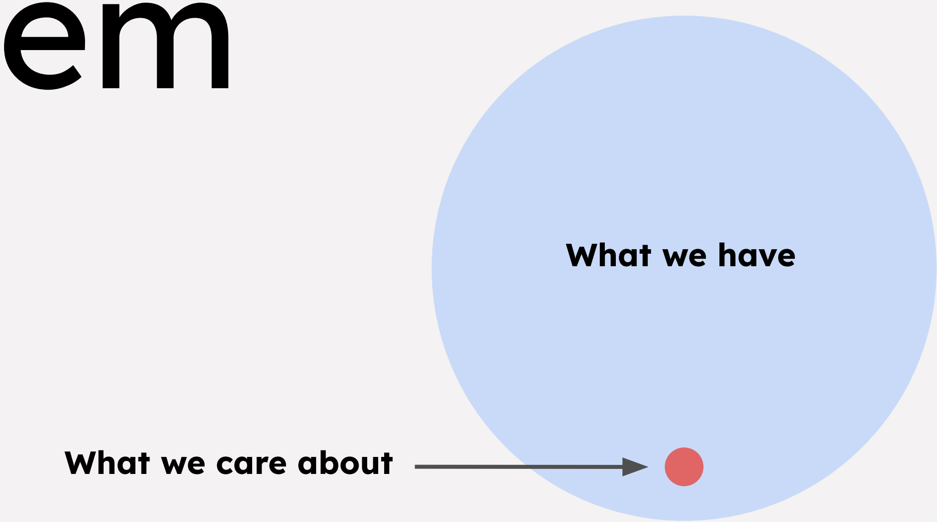
Brokers

Multi-messenger Instruments

Marshals

https://rubinobservatory.org/for-scientists/data-products/alerts-and-brokers

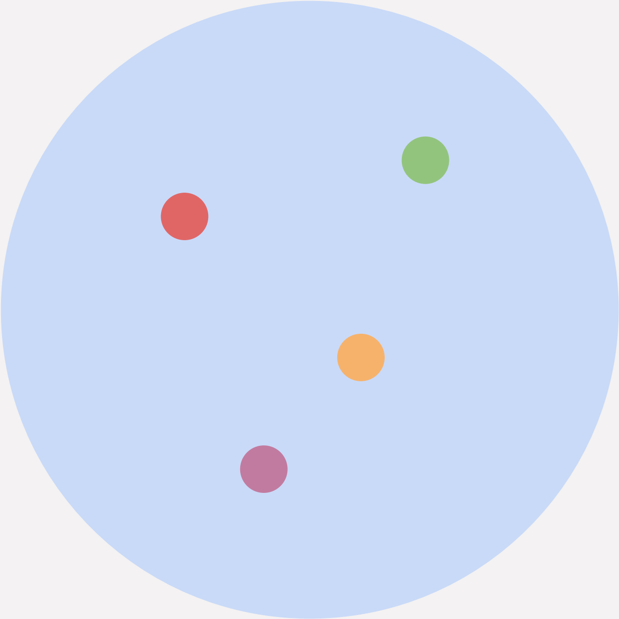
A ~~Astronomy~~ Big Data* Problem



*Rubin: 20 million alerts / night

An Astronomy Problem

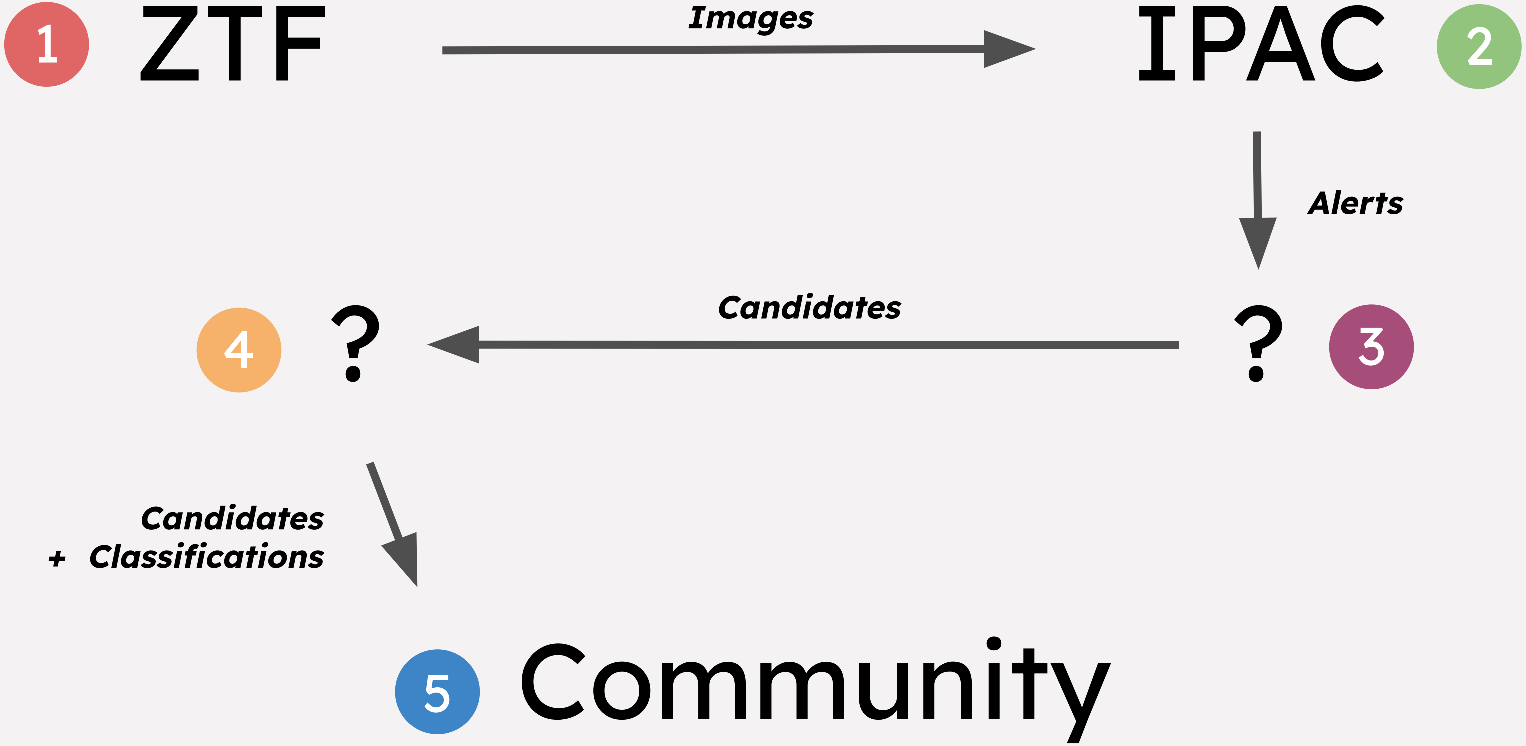
- What A cares about
- What B cares about
- What C cares about
- What D cares about



A latency Problem

- Rare objects
- Young objects
- Fast evolving objects
- ...

The best example of this (currently!), is ZTF



1 ZTF

Images

IPAC 2

Alerts

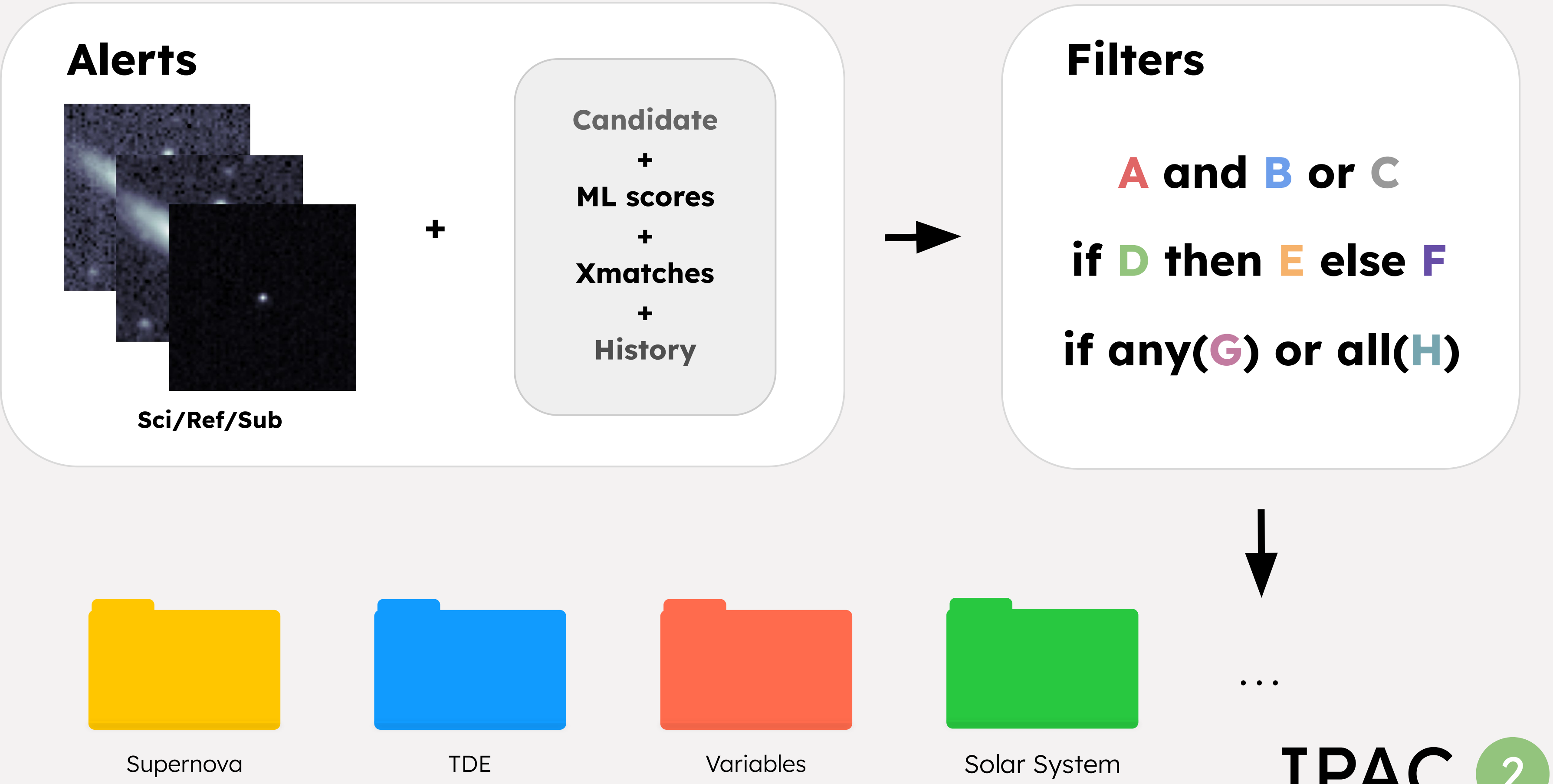
? 3

Candidates

4 ?

Candidates
+ *Classifications*

5 Community



Alerts

Sci/Ref/Sub

+

Candidate
+
ML scores
+
Xmatches
+
History

Filters

A and B or C

if D then E else F

if any(G) or all(H)

Supernova

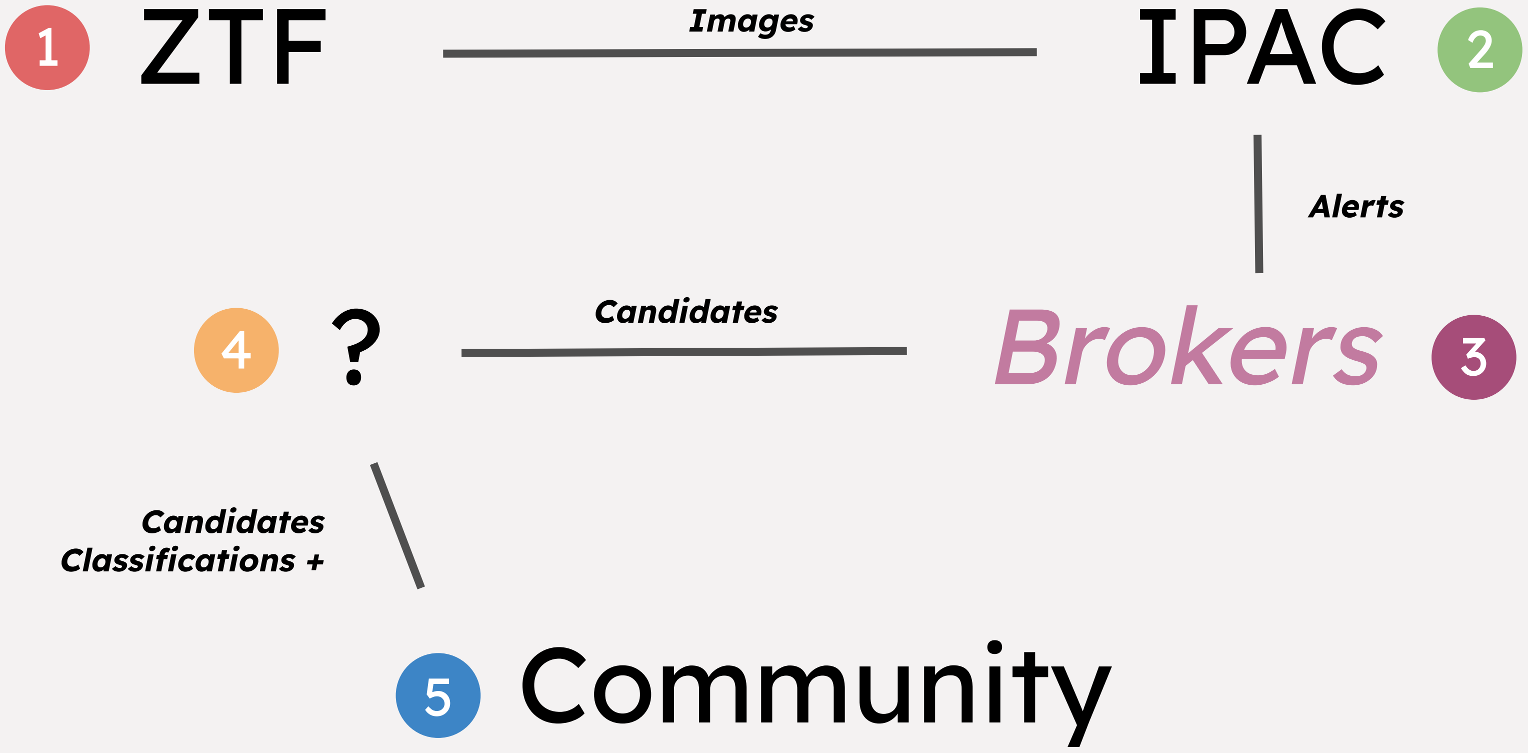
TDE

Variables

Solar System

...

IPAC 2



1 ZTF

Images

IPAC 2

Alerts

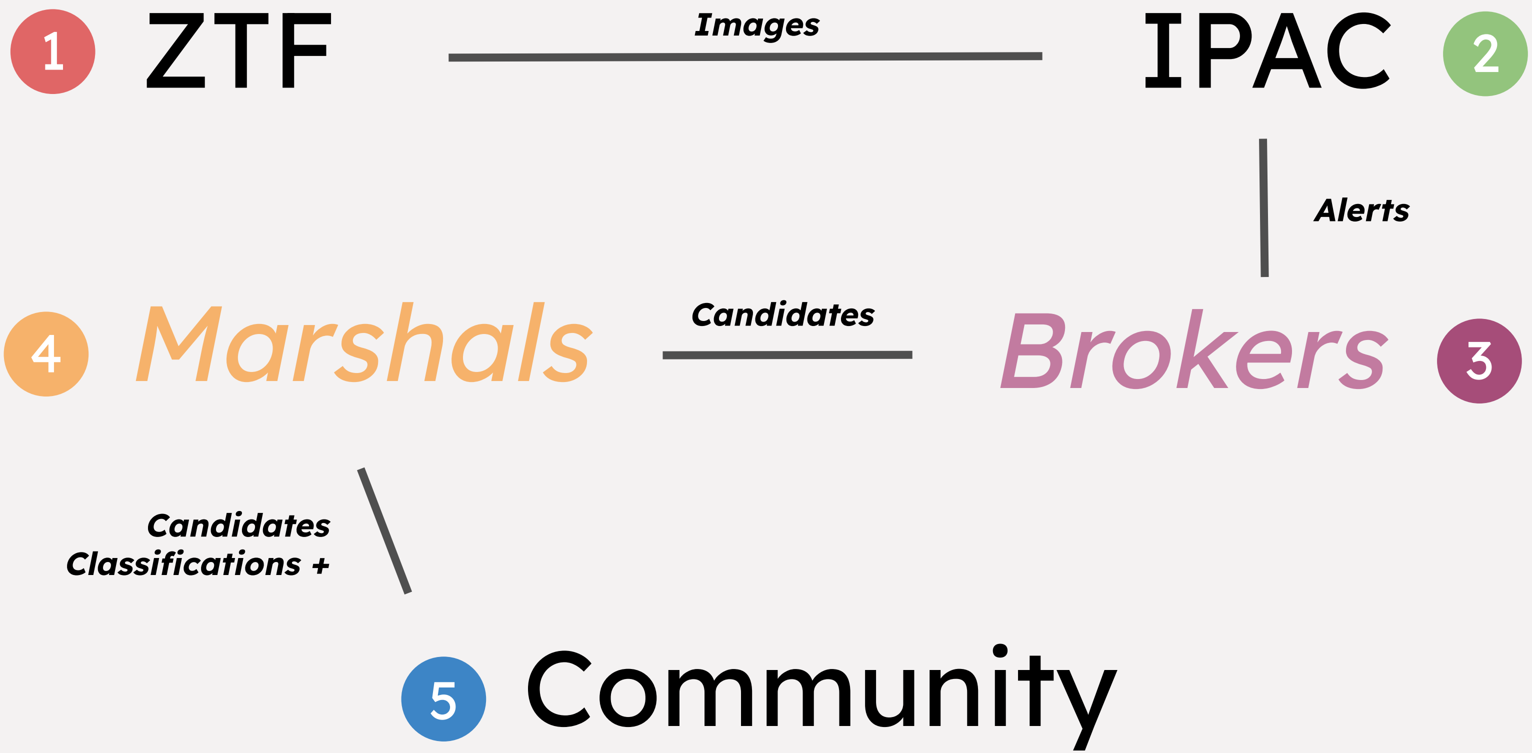
Brokers 3

Candidates

4 ?

Candidates
Classifications +

5 Community



1 ZTF

Images

IPAC 2

Alerts

3 Brokers

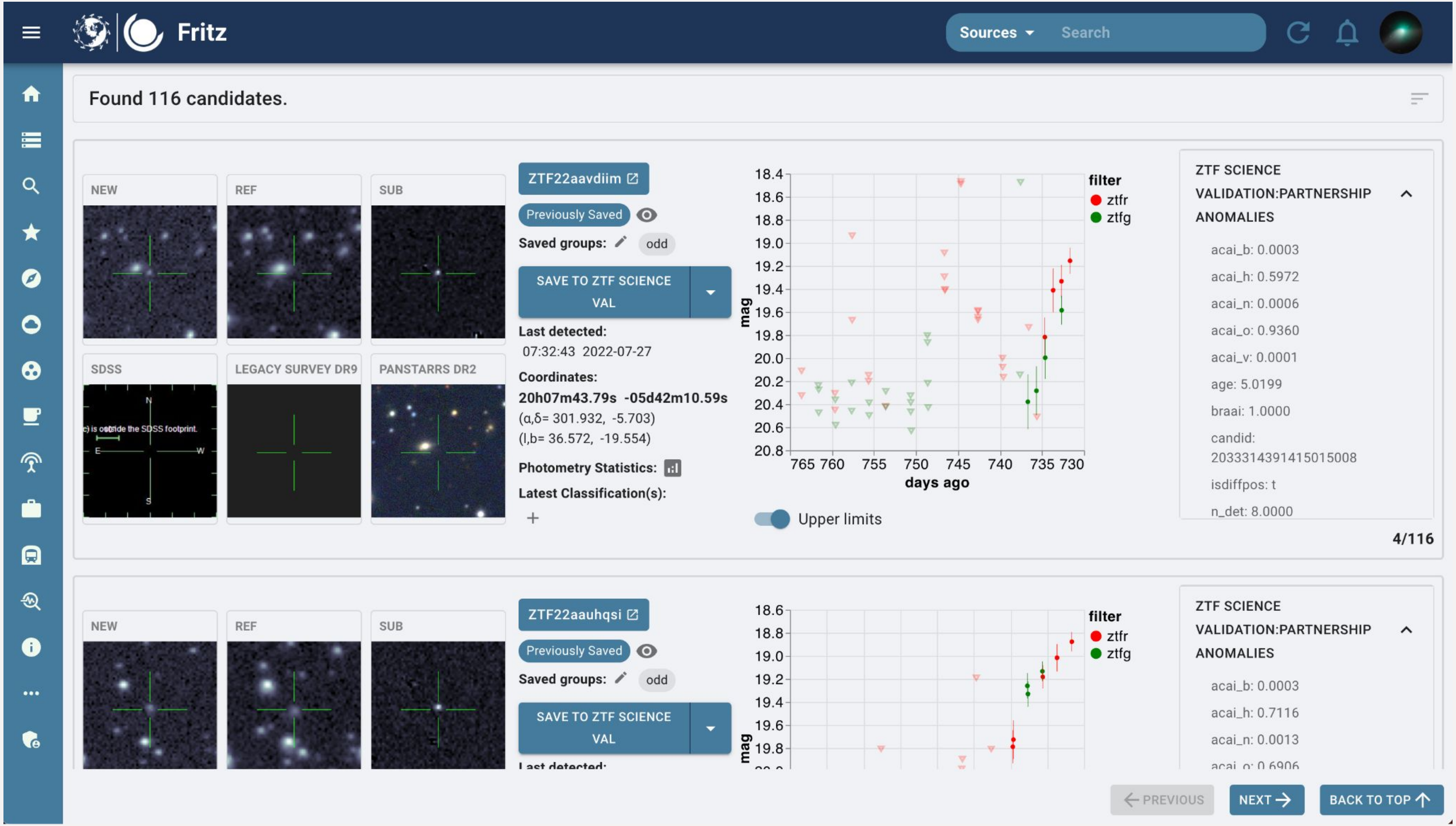
Candidates

4 *Marshals*

Candidates
Classifications +

5 Community

▶ BOOM



Found 116 candidates.

ZTF22aavdiim

Previously Saved

Saved groups: odd

SAVE TO ZTF SCIENCE VAL

Last detected:
07:32:43 2022-07-27

Coordinates:
20h07m43.79s -05d42m10.59s
(α,δ= 301.932, -5.703)
(l,b= 36.572, -19.554)

Photometry Statistics:

Latest Classification(s):

c) is outside the SDSS footprint.

ZTF SCIENCE VALIDATION:PARTNERSHIP ANOMALIES

acai_b: 0.0003
acai_h: 0.5972
acai_n: 0.0006
acai_o: 0.9360
acai_v: 0.0001
age: 5.0199
braai: 1.0000
candid: 2033314391415015008
isdiffpos: t
n_det: 8.0000

4/116

ZTF22aauhqsi

Previously Saved

Saved groups: odd

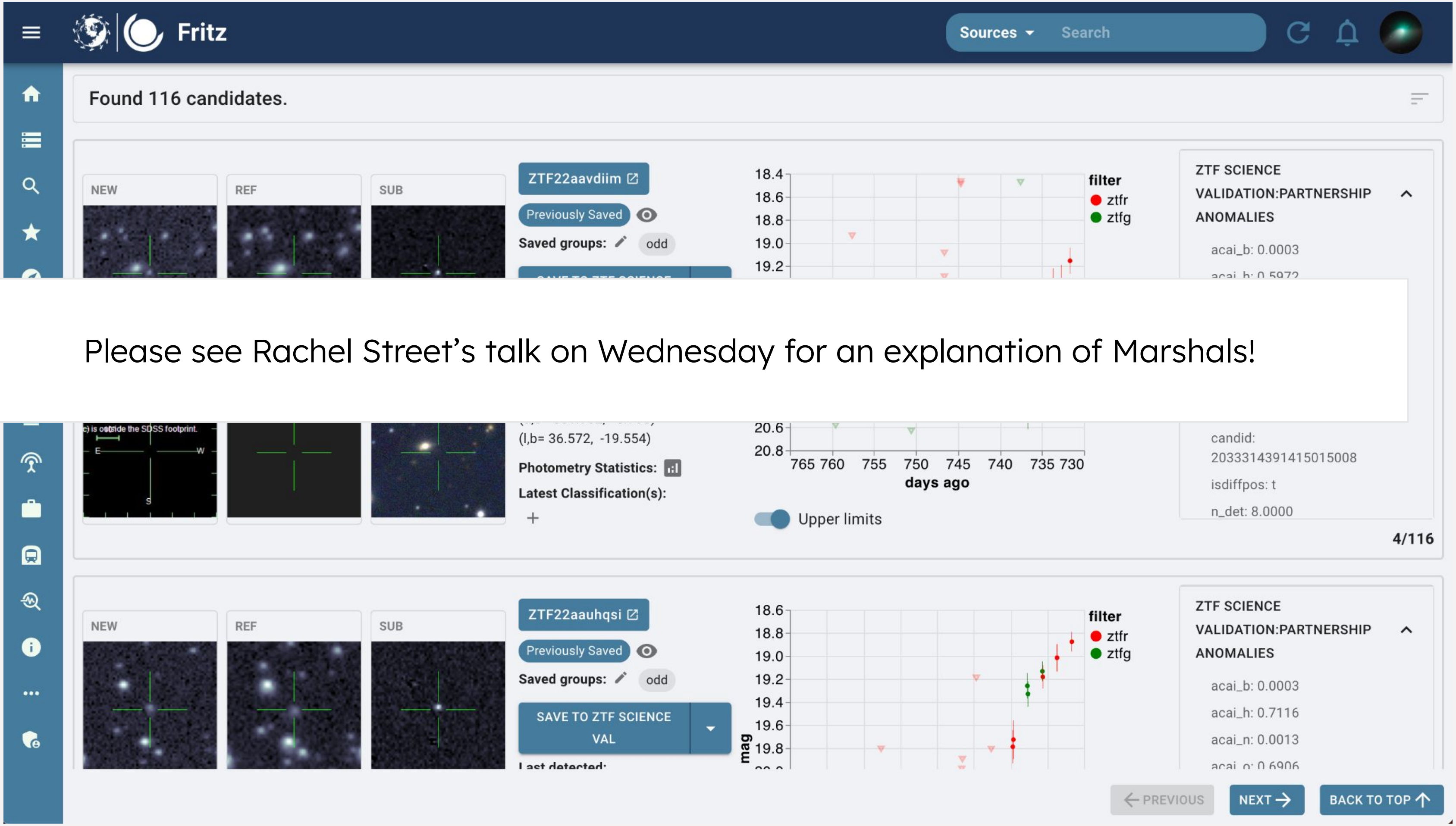
SAVE TO ZTF SCIENCE VAL

Last detected:

ZTF SCIENCE VALIDATION:PARTNERSHIP ANOMALIES

acai_b: 0.0003
acai_h: 0.7116
acai_n: 0.0013
acai_o: 0.6906

▶ BOOM



Fritz

Sources ▾ Search

Found 116 candidates.

NEW REF SUB

ZTF22aavdiim

Previously Saved

Saved groups: odd

ZTF SCIENCE VALIDATION:PARTNERSHIP ANOMALIES

acai_b: 0.0003

(l,b= 36.572, -19.554)

Photometry Statistics:

Latest Classification(s):

Upper limits

candid: 2033314391415015008

isdiffpos: t

n_det: 8.0000

4/116

NEW REF SUB

ZTF22aauhqsi

Previously Saved

Saved groups: odd

SAVE TO ZTF SCIENCE VAL

ZTF SCIENCE VALIDATION:PARTNERSHIP ANOMALIES

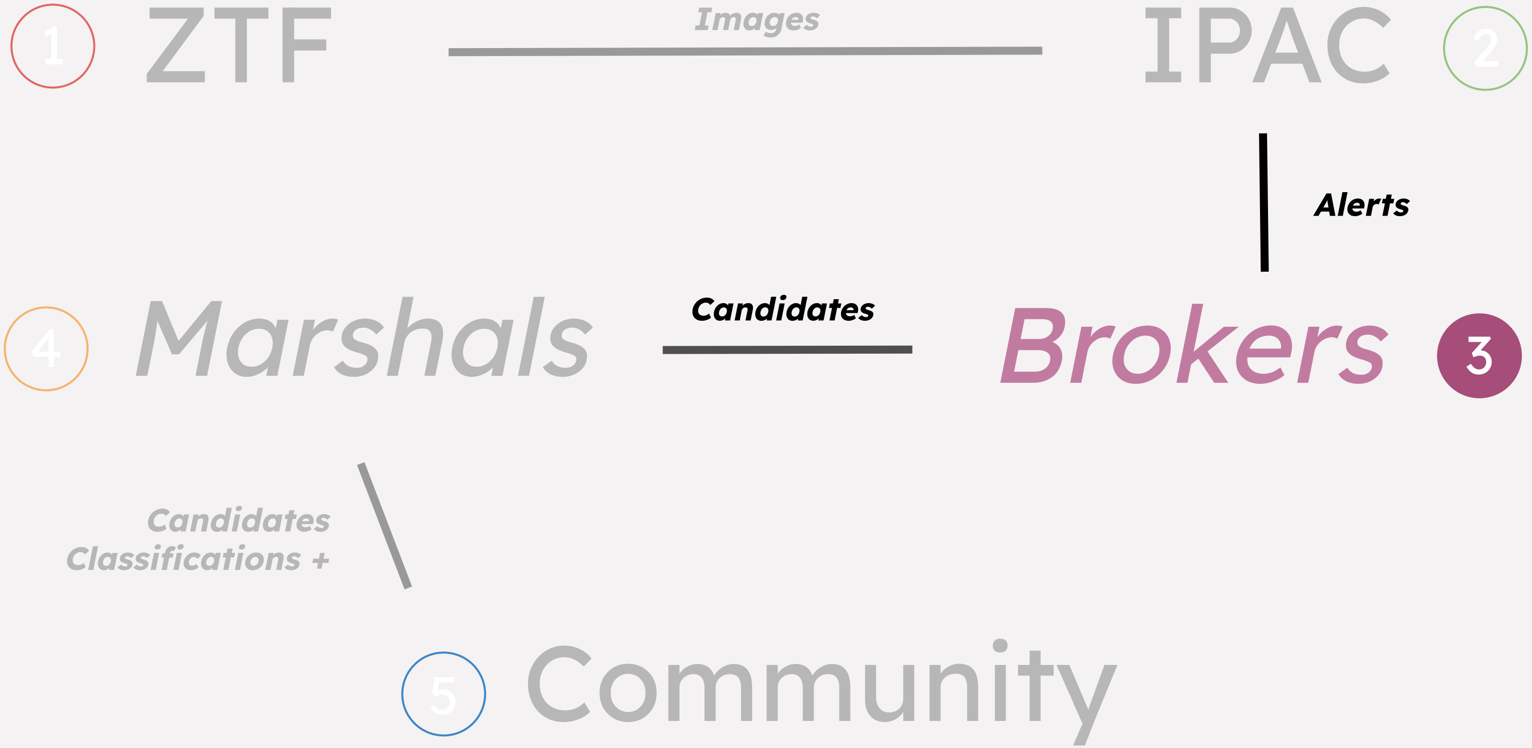
acai_b: 0.0003

acai_h: 0.7116

acai_n: 0.0013

acai_o: 0.6906

Please see Rachel Street’s talk on Wednesday for an explanation of Marshals!



1 ZTF

Images

IPAC 2

Alerts

4 Marshals

Candidates

Brokers 3

Candidates
Classifications +

5 Community

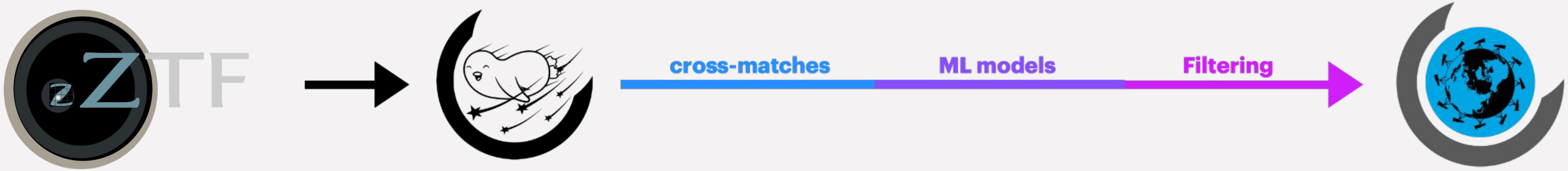
For BOOM we focus on the brokering side of things, though we should keep in mind what comes before and after it, as it dictates what the input and output should look like.

We want:

- **Flexible** - work for many science cases, support multiple instruments
- **Performant** - real time discoveries, little to no latency
- **Persistent** - archival searches
- **Reliable** - trusted by science users
- **Simple** - no user hardware required
- **Modular** - easily add new features

Currently: Kowalski

- Receive alerts from ZTF
- Add spatial cross-matches with archival catalogs + other streams*
- Update the full light curve for each object
- Run ML models on the alerts, such as **BTSbot**
- Run **user-defined filters** & **send the filtered data to SkyPortal/Fritz**



cross-matches

ML models

Filtering

But…

- **Volume:** LSST will produce x10 to x20 times more alerts than ZTF. With Kowalski, we already hit a hard limit and can't keep up when we reach the 1M alerts per night, for one survey.
- **Constant xmatching:** We want objects from each surveys to know about what objects in other surveys they match with (vs. the one-time xmatching we can do when we use archival catalogs).
- **Easier filter building:** the current system isn't user friendly, created a lot of frustration and significantly slows down / limits what we can do with Kowalski

BOOM

- **Multi-survey:** process the ***ZTF + Rubin*** alert streams, with ***continuous cross-match*** to use their ***combined light curves*** at filtering time.
- **Data enrichment:** Value added using ML scores using new multimodal models (Apple CiDeR), and xmatches with multiple static catalogs.
- **High performance:** built in ***Rust*** for speed, on top of a non-relational database w/ ***native spatial querying*** support. Processing divided in multiple tiers/types of “workers”, to ***scale operations independently and horizontally.***
- **User-defined:** **Filters are 100% customizable by users** through the SkyPortal marshal, output sent their for manual and automated vetting+triggering.

babamul

- **Built on top of BOOM: Built on BOOM's architecture** and alert enrichment, it is an **LSST filtering system.**
- **Public filters:** serving a set of **static filters** (based on transient/variable properties) **based on 7+ years of ZTF operations**
- **Low-latency:** Filter results **served over kafka**, along **with tools** for the clients **to filter further** (e.g. local ML, custom catalog xmatch, …).
- **Multi-survey information:** will include ZTF xmatch information, will include many catalog xmatch information, ML from AppleCiDEr

Automated Discoveries

- **Transient follow-up & classification**
- **MMA event ToOs for GW/neutrinos**
- **Enabled by BOOM + SkyPortal**
- **All of the above, 100% automatic**

AppleCiDEr

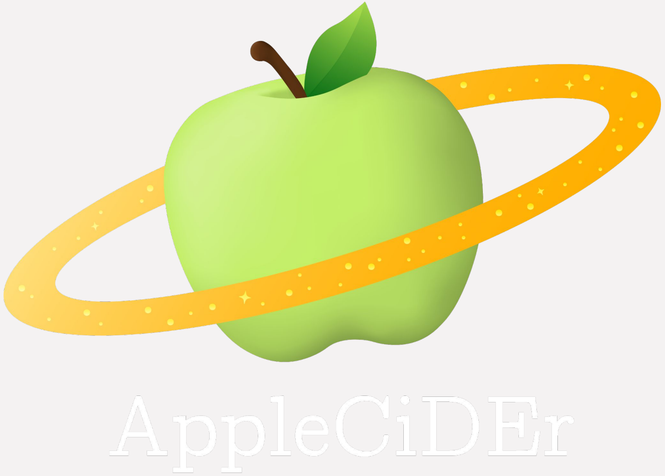
Multimodal ML framework: photometry, image cutouts, metadata, spectra

Photometry: [CLS]-Transformer 87.8% accuracy (59.2% using the Informer)

Images+Meta: AstroMiNN slightly better performance compared to BTSbot

Spectra: SpectraNeXt-2D 87.6% accuracy (83.7% using the GalSpecNet)

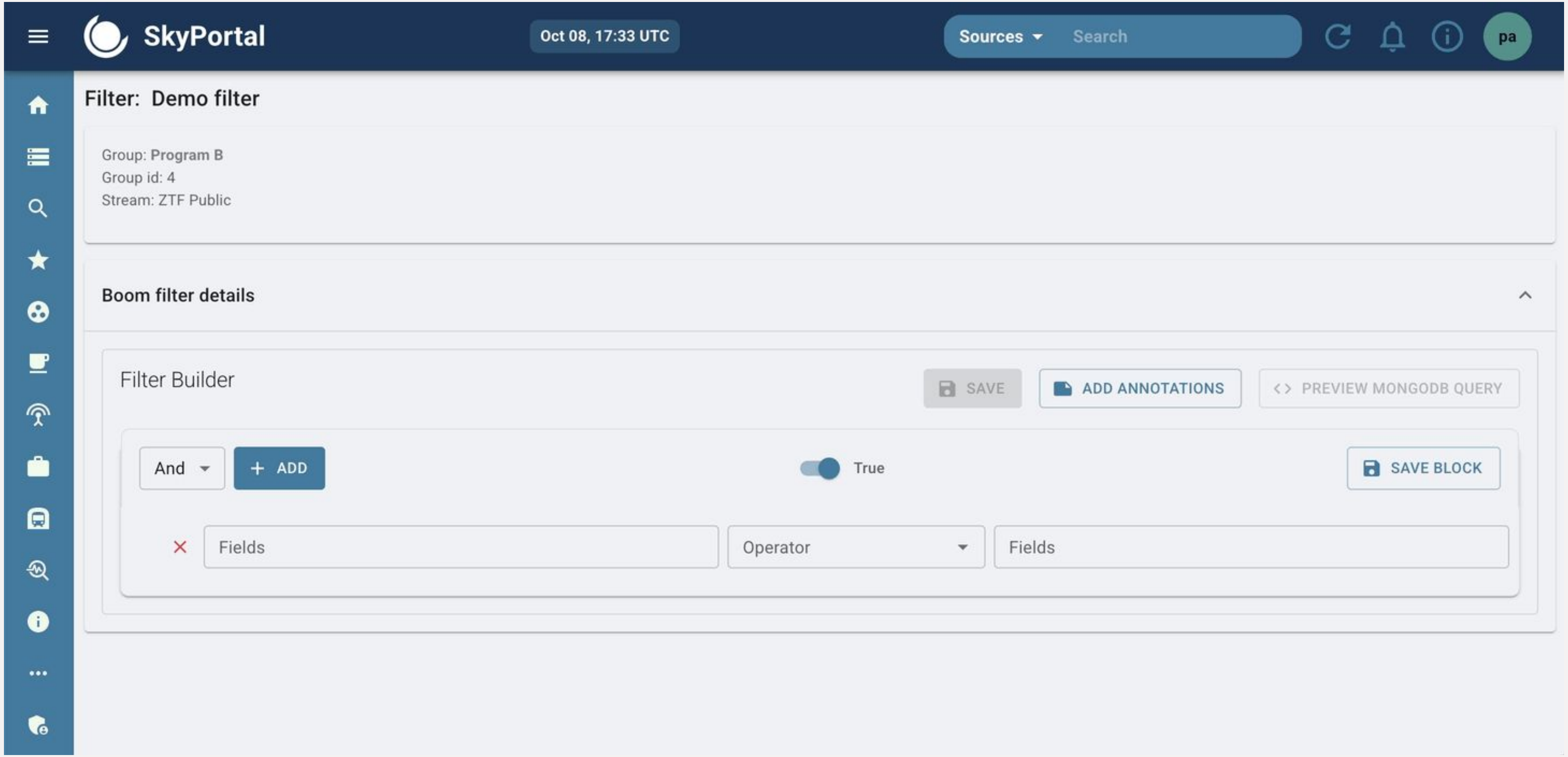
AppleCiDEr: Averages the results of the above networks. Almost ready for production!



Interested? Please read Junell et al. 2025 (2507.16088)

- **What about Babamul?**
  - Babamul is the **public-facing implementation of BOOM.** It is a publicly accessible LSST alert broker, that will **serve a number of static phenomenological filters that inherit from +7 years of experience looking for transients in the ZTF** public, partnership, and caltech streams
  - Instead of all alerts flowing to one topic (like the user-defined filters specifically designed to integrate with a marshal), **each filter will** be disjoint (where applicable/relevant, to avoid duplication) and **have a corresponding kafka topic.** Users will be provided with a babamul documentation web-page where they can pick which topics/features they are interested in, and have **code to read from the streams auto-generated** (ala GCN).
  - Ultimately, we'd like to serve a number of **community-driven modules** that sit after reading from Kafka and before outputting the data to other systems, t**o subsample further on the incoming streams**.

A Filter Interface



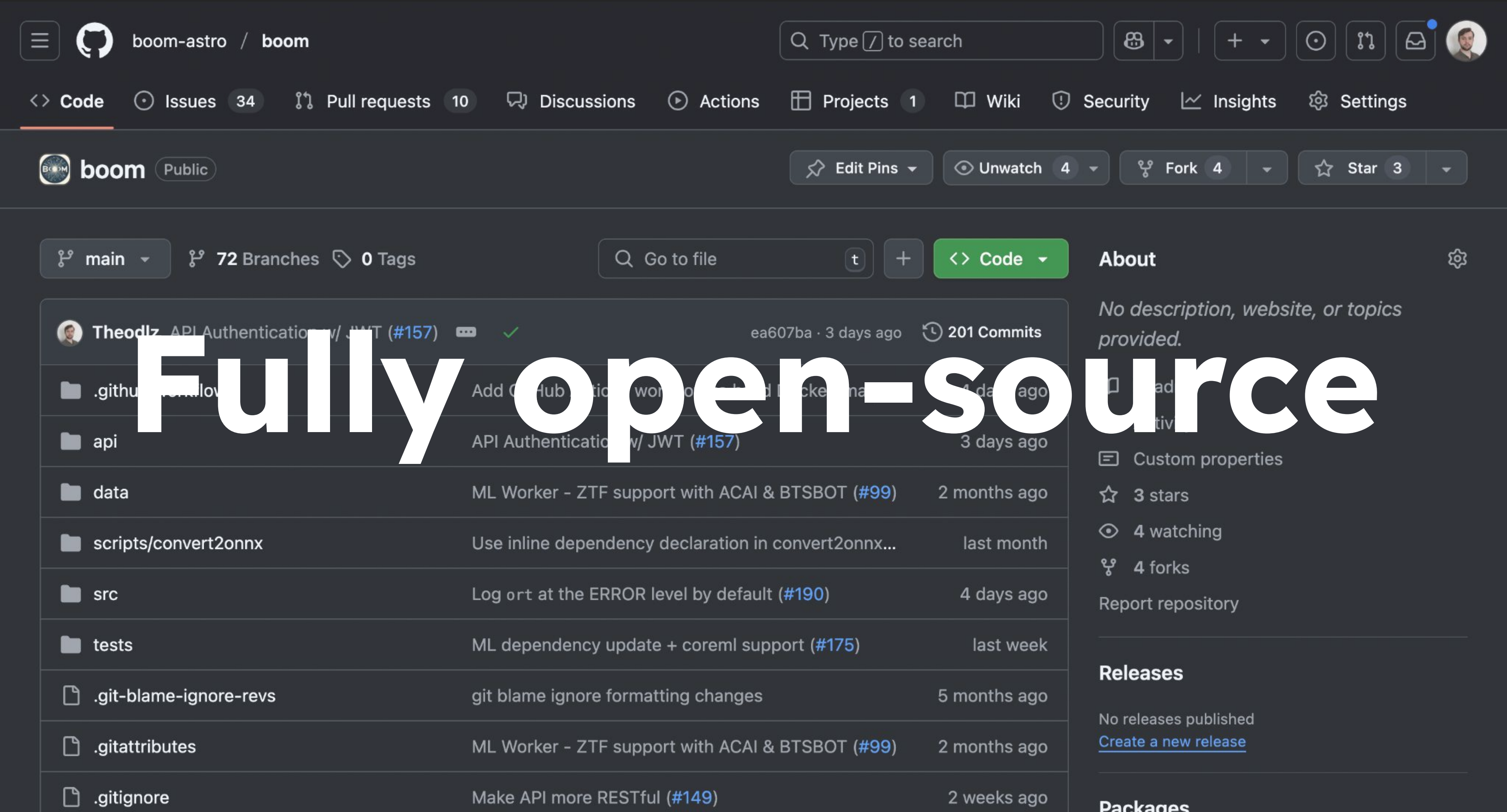
Filter: Demo filter

Group: Program B
Group id: 4
Stream: ZTF Public

Boom filter details

Filter Builder

Fully open-source



boom-astro / boom

Code · Issues 34 · Pull requests 10 · Discussions · Actions · Projects 1 · Wiki · Security · Insights · Settings

boom Public

Edit Pins · Unwatch 4 · Fork 4 · Star 3

main · 72 Branches · 0 Tags · Go to file · Code

Theodlz API Authentication w/ JWT (#157) ✓ ea607ba · 3 days ago · 201 Commits

.github/workflows	Add GitHub … workflow … Docker …	4 days ago
api	API Authentication w/ JWT (#157)	3 days ago
data	ML Worker - ZTF support with ACAI & BTSBOT (#99)	2 months ago
scripts/convert2onnx	Use inline dependency declaration in convert2onnx...	last month
src	Log `ort` at the ERROR level by default (#190)	4 days ago
tests	ML dependency update + coreml support (#175)	last week
.git-blame-ignore-revs	git blame ignore formatting changes	5 months ago
.gitattributes	ML Worker - ZTF support with ACAI & BTSBOT (#99)	2 months ago
.gitignore	Make API more RESTful (#149)	2 weeks ago

About

No description, website, or topics provided.

Custom properties

3 stars

4 watching

4 forks

Report repository

Releases

No releases published

Create a new release

Packages

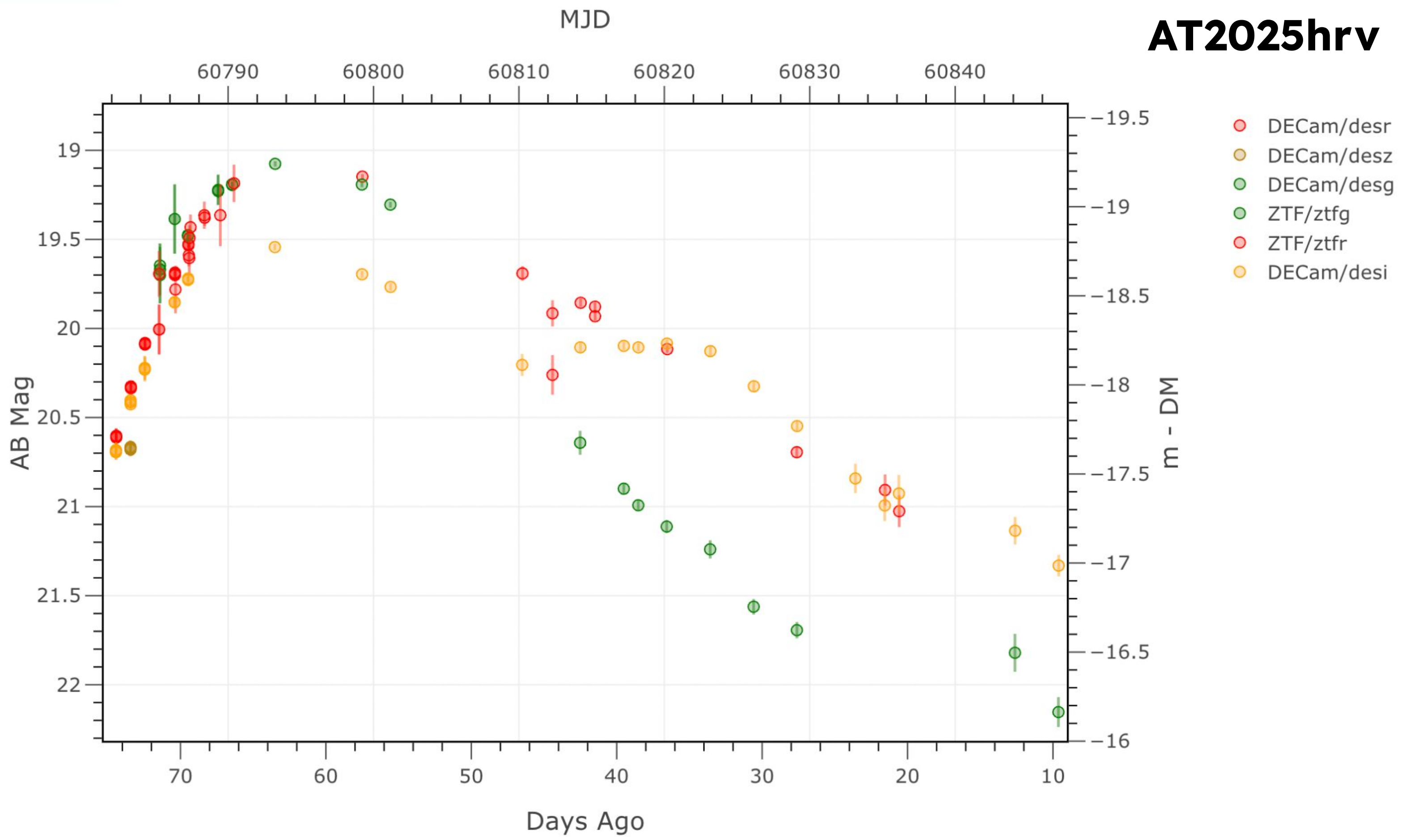
Thanks for Listening!

Where are we in the development?

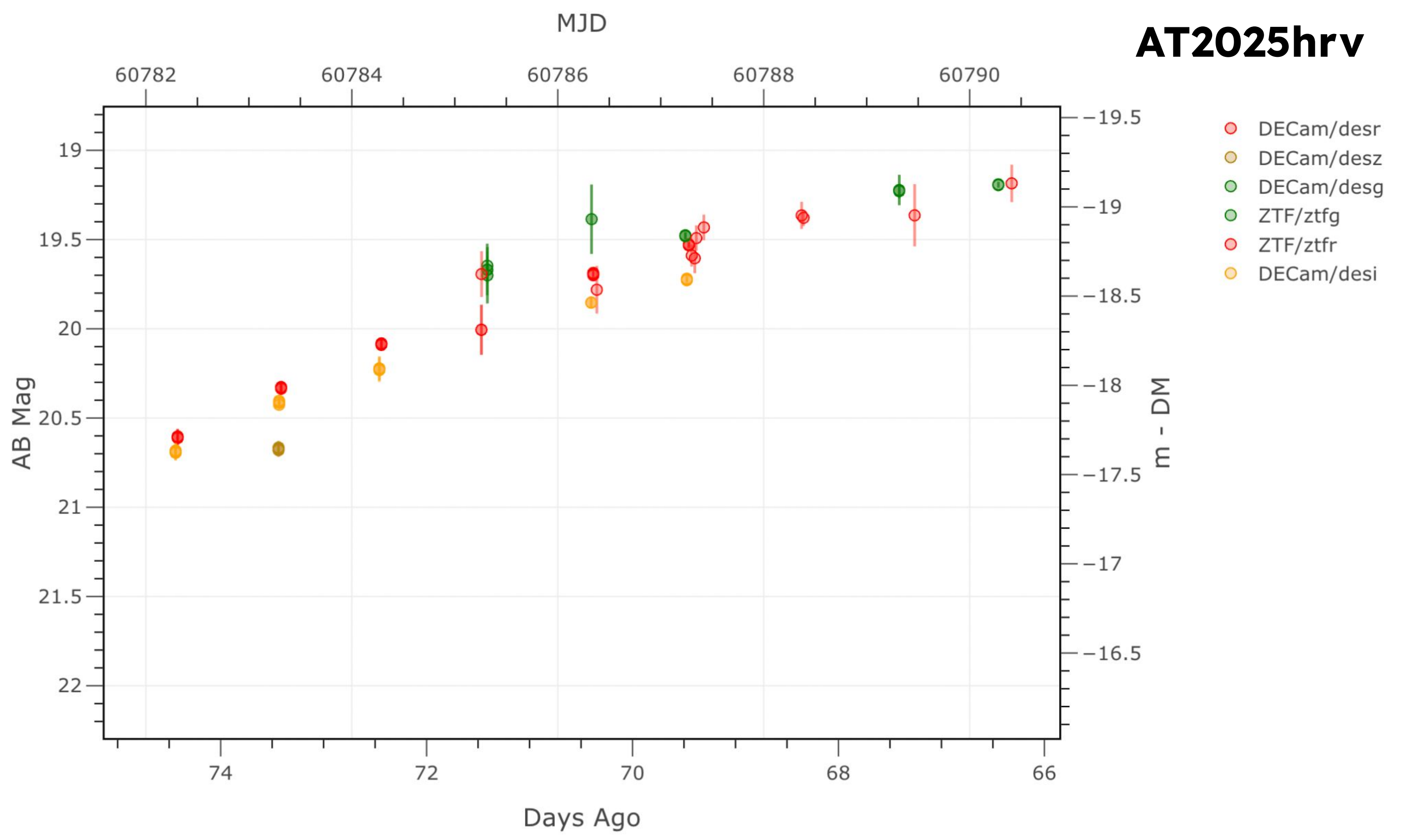
- **Multi-survey support:**
  - **We can successfully process the ZTF and Rubin alert stream**. Each worker (except ML as we do not have models for Rubin yet) has a generic “trait” with as many reusable methods as possible, and we implement that “trait” for each survey, where we can have the survey-specific processing steps.
  - **Every new ZTF alerts is cross-matched with the LSST** alert-based **object collection,** simply based on position (closest is best). *In future iterations (as we get real data to play with), we want to explore xmatching as a function of position, time, and brightness.*

- **Multi-survey support:**
    - We’ve **successfully used real DECAM alerts** (from last month’s ZTF+DECAM experiment, an amazing **opportunity to test multi-stream operations such as crossmatching.**
    - Also **developed multi-survey filtering** in this context, where filters running on the ZTF alert stream can required a coincidence with DECAM objects, and make use of DECAM lightcurves if any.
    - **It gaves us a preview of what we’ll be able to do w/ LSST, as we found a number of ZTF objects which on their first ever detection, had a fainter DECAM pre-detection.** Ruling them out as moving objects and giving us constraints on their rate of evolution.

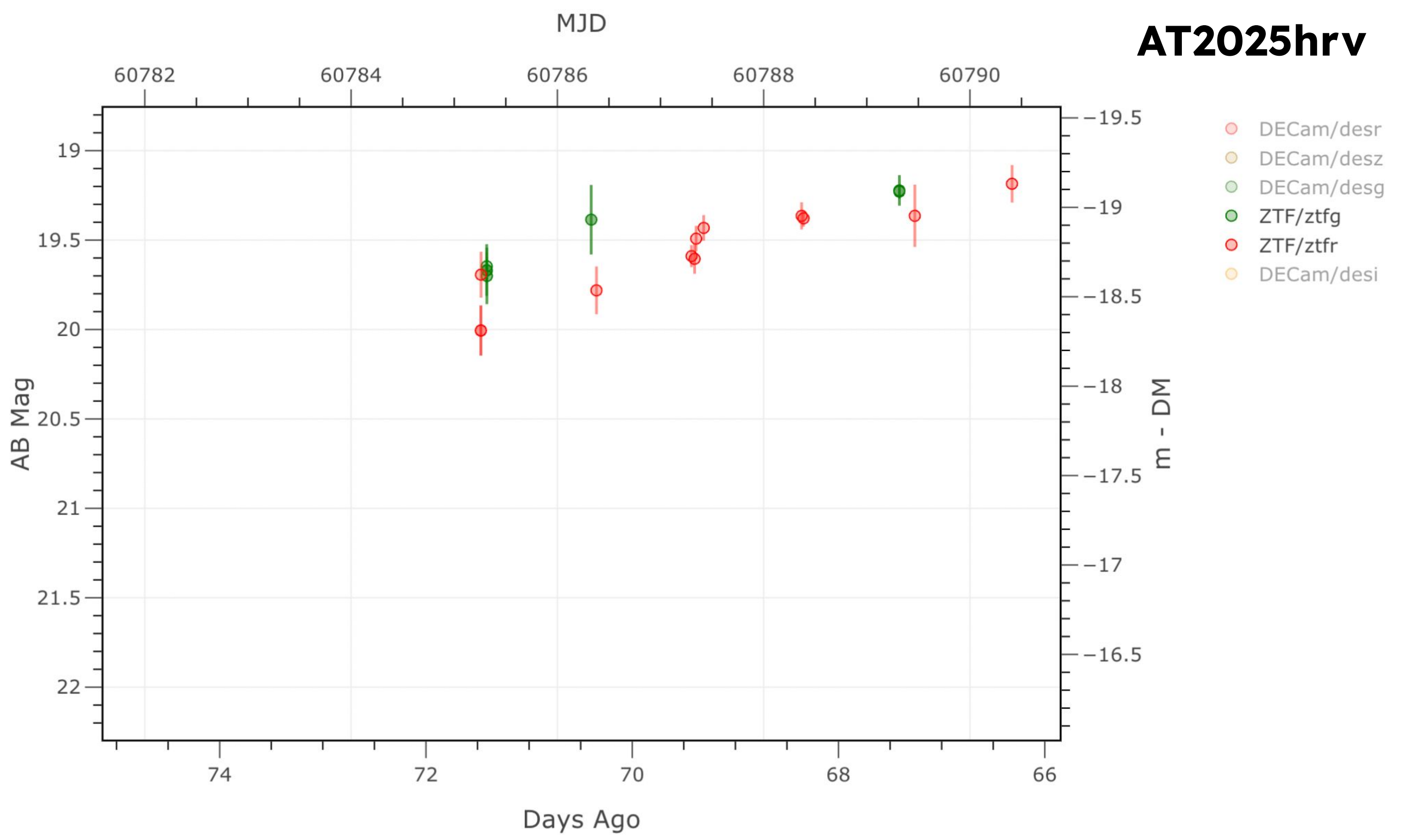
AT2025hrv

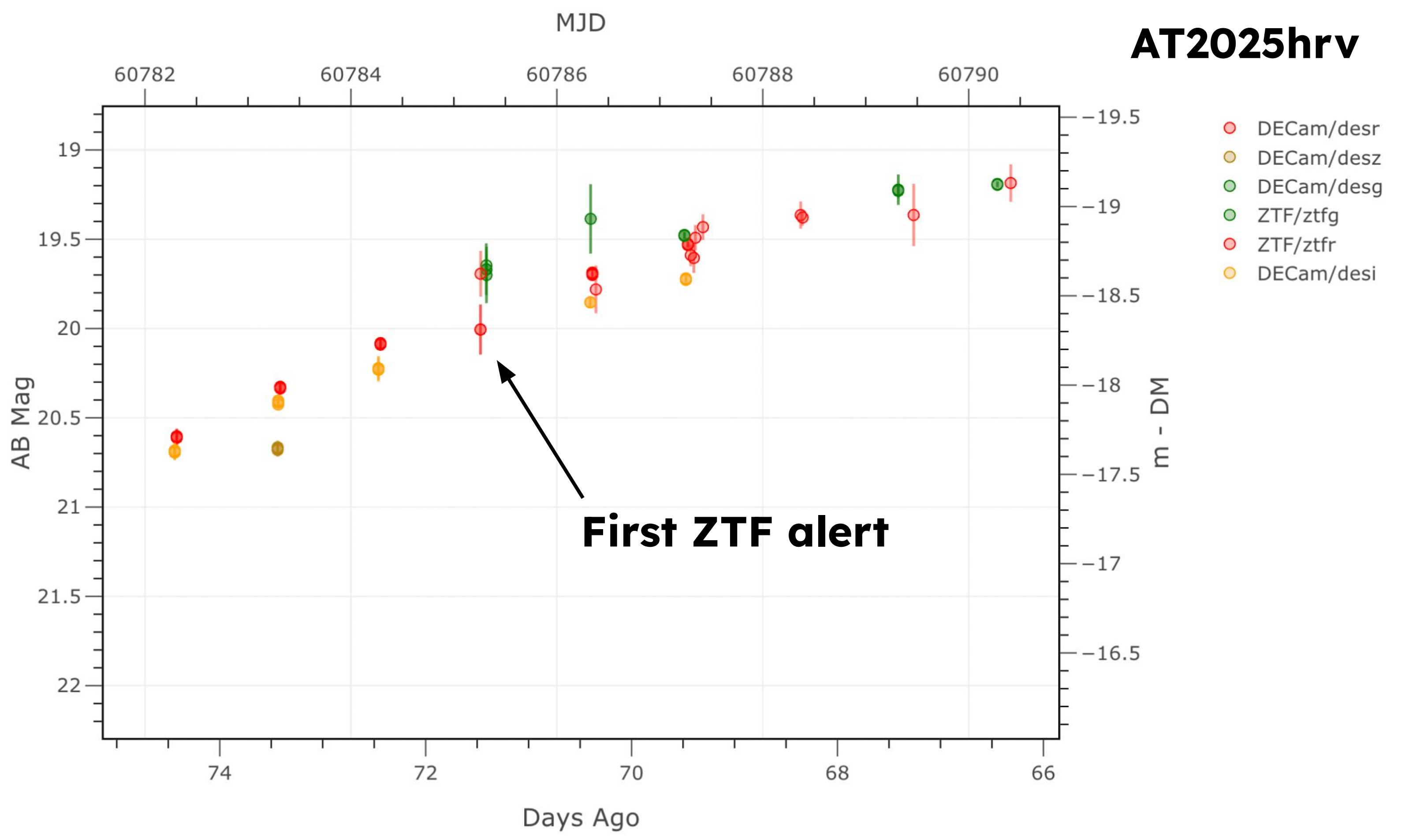

MJD
60790
60800
60810
60820
60830
60840
AB Mag
19
19.5
20
20.5
21
21.5
22
m - DM
−19.5
−19
−18.5
−18
−17.5
−17
−16.5
−16
70
60
50
40
30
20
10
Days Ago
DECam/desr
DECam/desz
DECam/desg
ZTF/ztfg
ZTF/ztfr
DECam/desi

AT2025hrv


MJD
60782
60784
60786
60788
60790
AB Mag
19
19.5
20
20.5
21
21.5
22
m - DM
−19.5
−19
−18.5
−18
−17.5
−17
−16.5
74
72
70
68
66
Days Ago
DECam/desr
DECam/desz
DECam/desg
ZTF/ztfg
ZTF/ztfr
DECam/desi

AT2025hrv

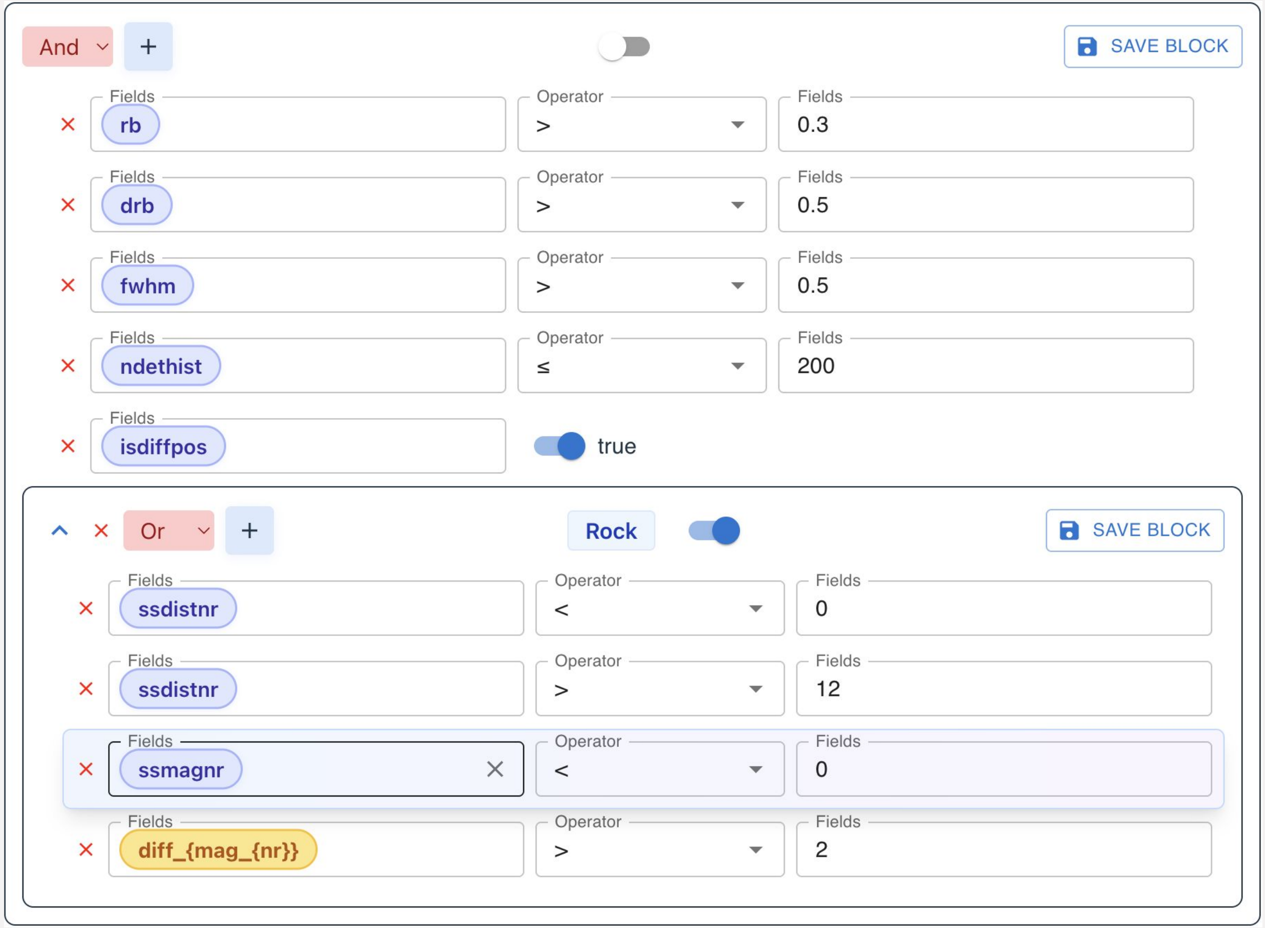

MJD
60782
60784
60786
60788
60790
19
19.5
20
20.5
21
21.5
22
AB Mag
−19.5
−19
−18.5
−18
−17.5
−17
−16.5
m - DM
74
72
70
68
66
Days Ago
DECam/desr
DECam/desz
DECam/desg
ZTF/ztfg
ZTF/ztfr
DECam/desi


MJD
60782
60784
60786
60788
60790
AT2025hrv
AB Mag
19
19.5
20
20.5
21
21.5
22
m - DM
−19.5
−19
−18.5
−18
−17.5
−17
−16.5
Days Ago
74
72
70
68
66
DECam/desr
DECam/desz
DECam/desg
ZTF/ztfg
ZTF/ztfr
DECam/desi
First ZTF alert

- **Multi-survey support:**
  - **A challenge is the heterogeneity of the data products**. For example, DECAM gives us forced aperture photometry while ZTF gives us history of past detections with PSF photometry and forced psf photometry, …
  - Aside from these fundamental differences between data products, their **schemas also differ** aside from some essential quantities (objectId, candid, ra, dec, time, band, …)
  - As we add more moving parts of the system and develop features related to multi-survey filtering, we discover where **schema differences still need to be resolved to facilitate building filters.**

- **Facilitating filter building:**
  - Most brokers offer one of:
    - static filters that users can then read from and implement their own filtering downstream
    - Ways to perform limited filtering from a web app
    - Ways to write your own Python filtering code and have the broker run it for you.
  - In Kowalski, we made use of the aggregation pipeline of MongoDB to let users write simple and complex pipelines that the broker ran for them. However, it is very clear that the amount of database-specific knowledge expected from them isn’t realistic, and made it a major blocker.

- **Facilitating filter building:**
  - **Developing a UI/web-based no-code tool**, **where filters can be composed without any database or programming language specific knowledge required.** The only requirement is to be familiar with the data product.
  - As users develop filtering “blocks” (sets of conditions) and custom variables they use in those, they **will be able to share** blocks **with other users,** that can simply import them in their own filter.
  - Overtime, as any given block/variable seems to be re-used across a significant number of filters, we can decide to **pre-compute common blocks** at alert ingestion time so simplify the filters further, and deduplicate operations.



rb > 0.3

drb > 0.5

fwhm > 0.5

ndethist ≤ 200

isdiffpos true

ssdistnr < 0

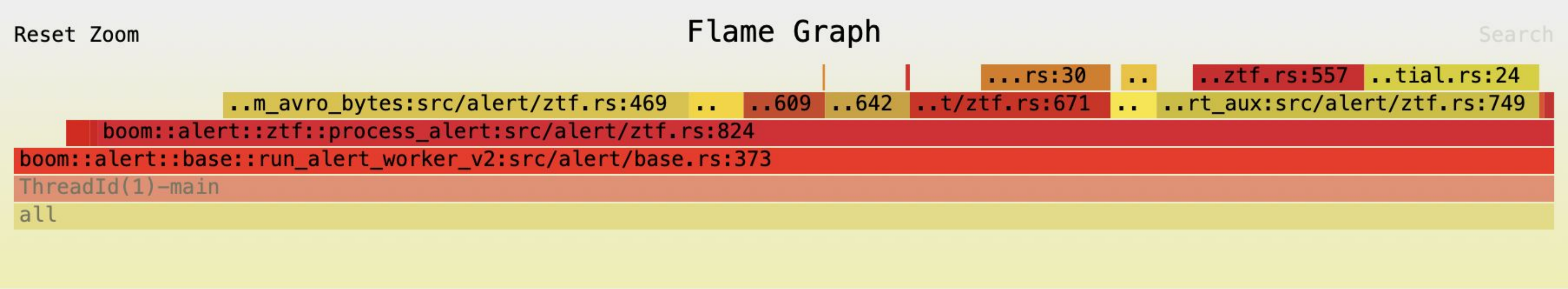
ssdistnr > 12

ssmagnr < 0

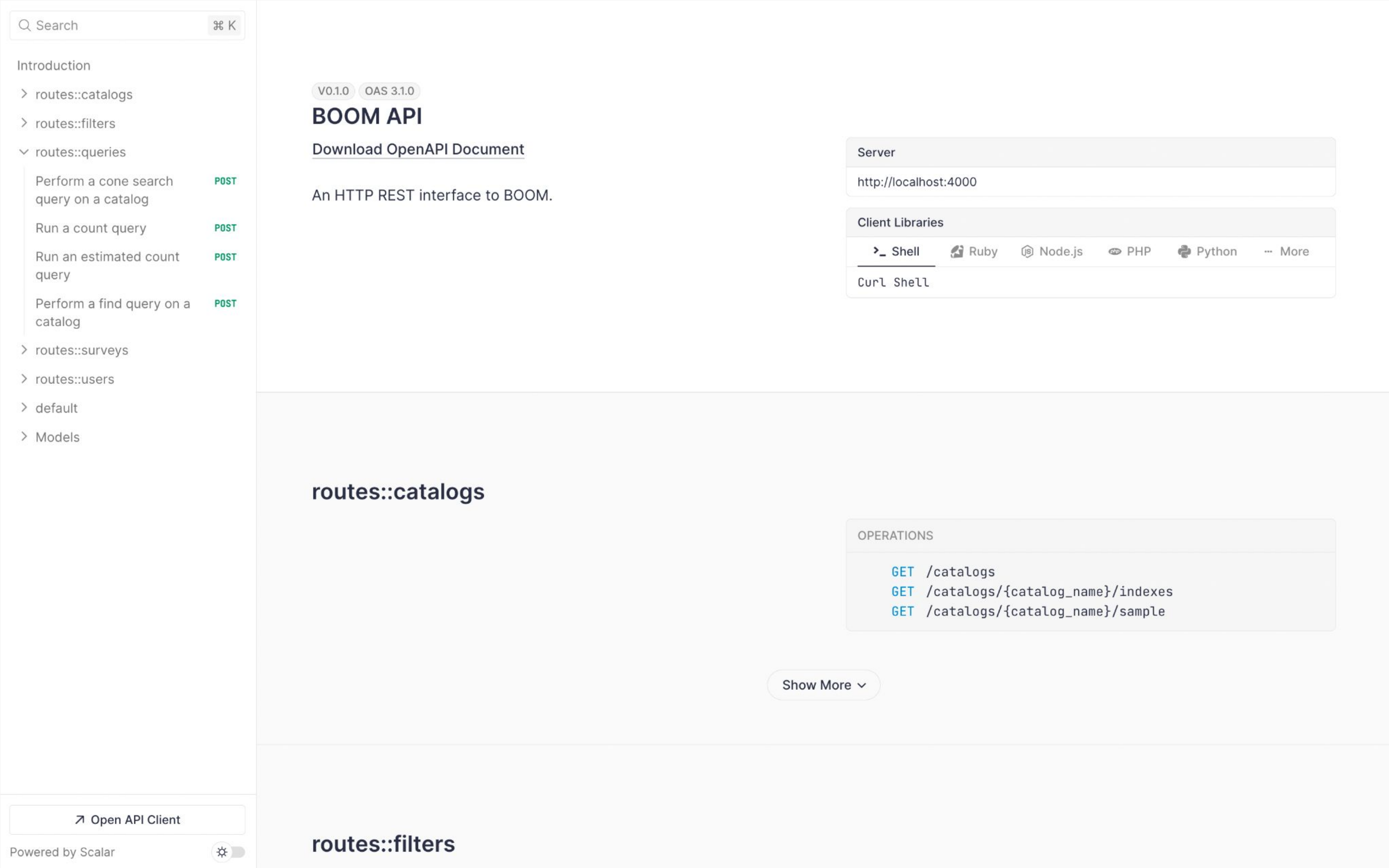
diff_{mag_{nr}} > 2

- **Integration with Marshals (SkyPortal)**
  - While BOOM doesn’t need to know about a marshal, it needs to know about BOOM.
  - Developing a **modular plugin system for SkyPortal,** to let anyone develop modules that can simply be referred to (pointer to their location on GitHub and version) and configured directly in SkyPortal’s config file, **without the need to extend the SkyPortal codebase itself.** Modules are registered (and compatibility assessed) at start-up.
  - A **dedicated BOOM module** has already been developed for ZTF, and can successfully populate any SkyPortal deployment with candidates identified by BOOM. Latency is much lower than the Kowalski + SkyPortal integration, as the overhead of multiple round trips to an HTTP API are avoided.

- **Observability:**
  - **Integrating the entire codebase with tracing/spanning,** so every single function can be measured and logged. Tracing comes at a cost so that should be accounted for, however logging levels can be decided at compile time, meaning the same codebase can have high levels of logging for debugging/benchmarking but lower levels for production, at 0 cost.
  - **Building visualization tools** such as *flame graphs* to measure where time is spent across the app.
  - **Building a benchmarking pipeline** to measure BOOM's performance against Kowalski's, based on *calkit.*


Reset Zoom
Flame Graph
Search
...rs:30
..
..ztf.rs:557
..tial.rs:24
..m_avro_bytes:src/alert/ztf.rs:469
..
..609
..642
..t/ztf.rs:671
..
..rt_aux:src/alert/ztf.rs:749
boom::alert::ztf::process_alert:src/alert/ztf.rs:824
boom::alert::base::run_alert_worker_v2:src/alert/base.rs:373
ThreadId(1)-main
all

- **HTTP API:**
  It sits on top of BOOM to **give access to the data after it is processed** (for archival searches **or semi-real-time** operations that lets users implement highly customized pipelines) and to **upload/update user-defined filters**. A lot of progress has been made recently, but:
  - It needs to be tested very thoroughly with unit tests, for successful queries but also for all expected failure modes.
  - It will be edited to accommodate the Filter Building UI that Thomas Culino (CIT) is developing.
  - It would benefit - just like Kowalski did - from an **API client** in commonly used programming languages such as Python.
  - It is **already documented**, but while the documentation is hosted alongside the server, we should have it published on a static web page ala GitHub pages.



BOOM API

An HTTP REST interface to BOOM.

Server

http://localhost:4000

Client Libraries

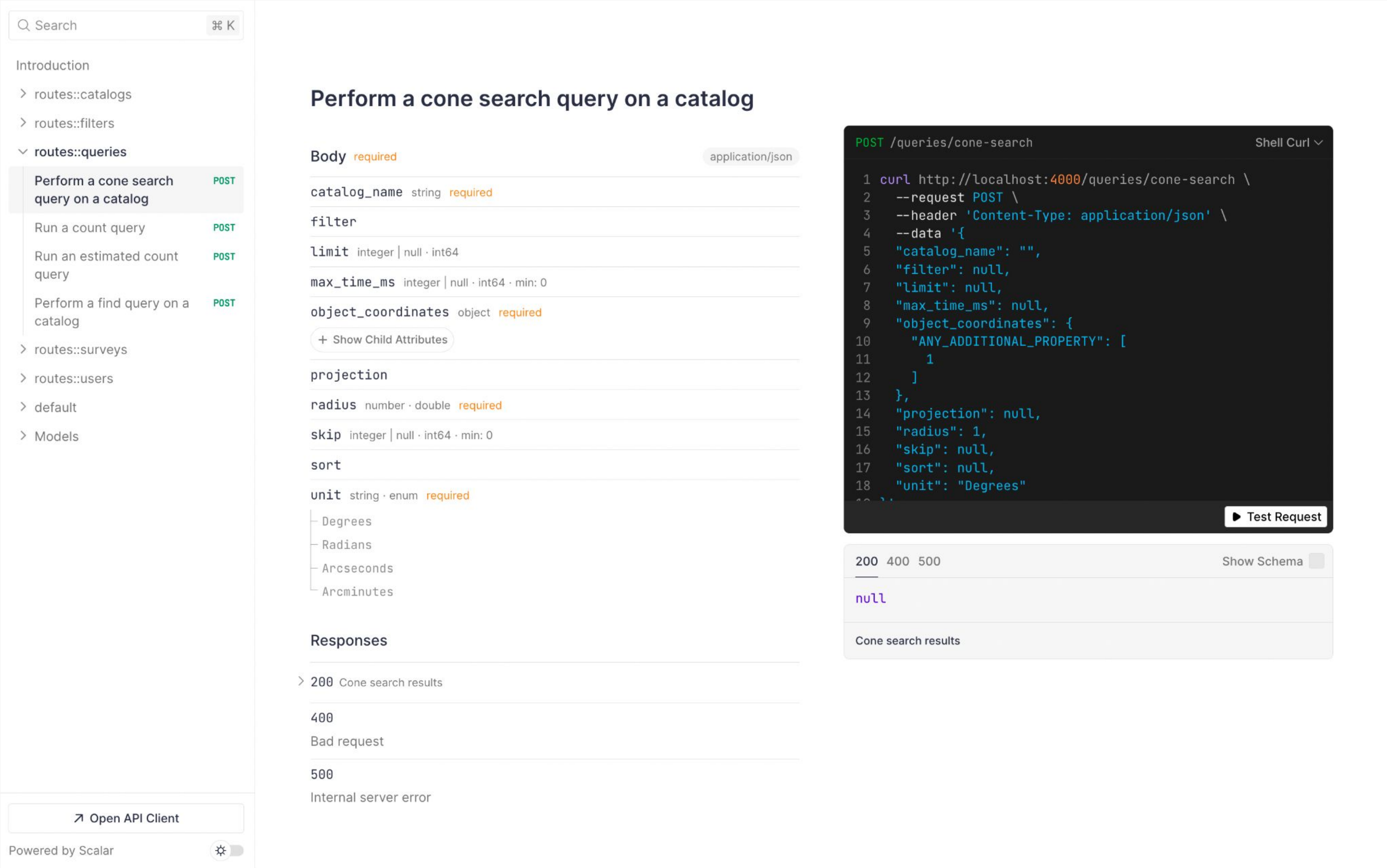
Curl Shell

routes::catalogs

OPERATIONS

```
GET  /catalogs
GET  /catalogs/{catalog_name}/indexes
GET  /catalogs/{catalog_name}/sample
```

routes::filters



Perform a cone search query on a catalog

Body required application/json

`catalog_name` string required

`filter`

`limit` integer | null · int64

`max_time_ms` integer | null · int64 · min: 0

`object_coordinates` object required

+ Show Child Attributes

`projection`

`radius` number · double required

`skip` integer | null · int64 · min: 0

`sort`

`unit` string · enum required

- Degrees
- Radians
- Arcseconds
- Arcminutes

Responses

200 Cone search results

400
Bad request

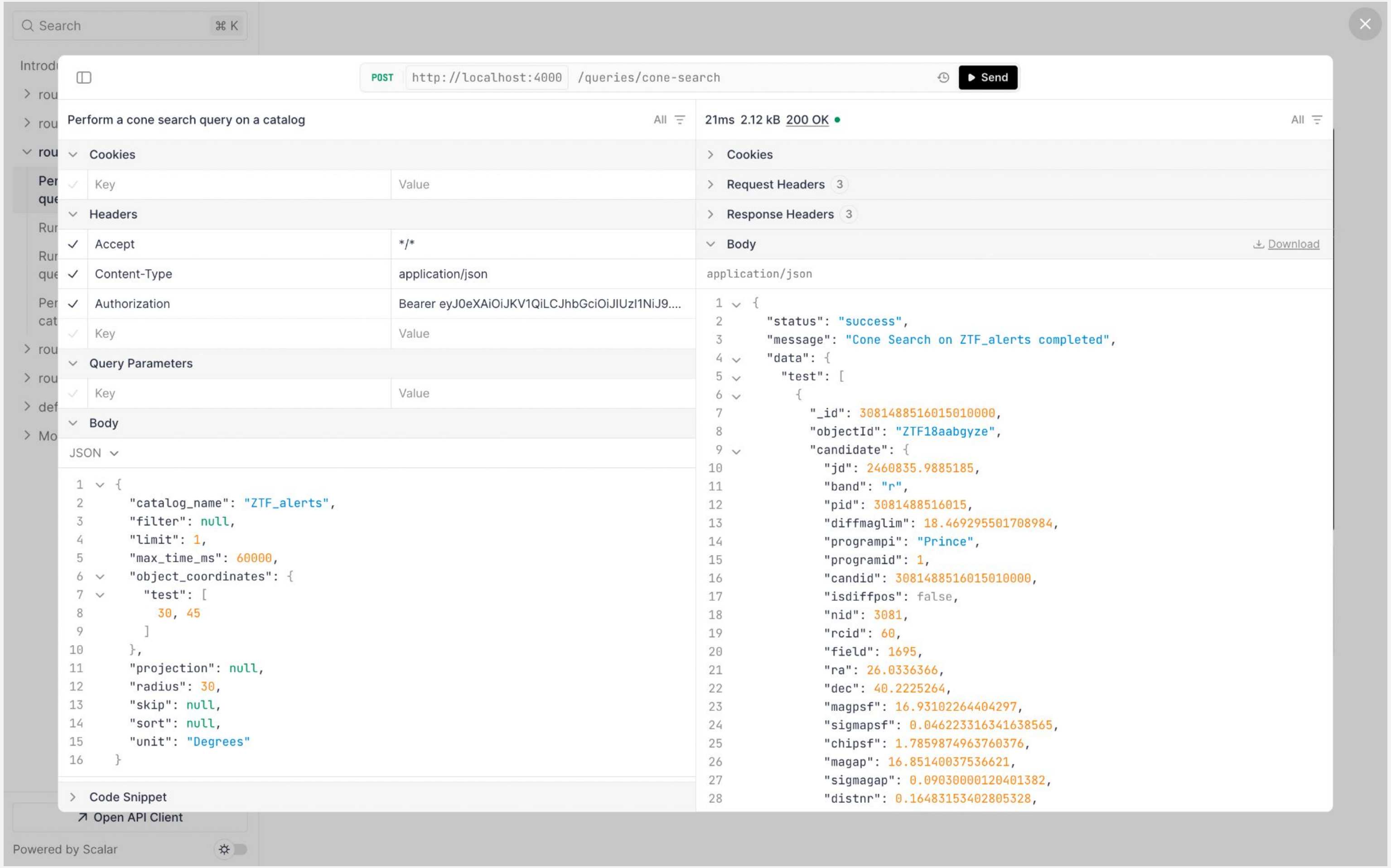
500
Internal server error

POST /queries/cone-search Shell Curl

```
curl http://localhost:4000/queries/cone-search \
  --request POST \
  --header 'Content-Type: application/json' \
  --data '{
  "catalog_name": "",
  "filter": null,
  "limit": null,
  "max_time_ms": null,
  "object_coordinates": {
    "ANY_ADDITIONAL_PROPERTY": [
      1
    ]
  },
  "projection": null,
  "radius": 1,
  "skip": null,
  "sort": null,
  "unit": "Degrees"
```

```
null
```

Cone search results



POST http://localhost:4000 /queries/cone-search

Perform a cone search query on a catalog

Key	Value
Accept	*/*
Content-Type	application/json
Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9....

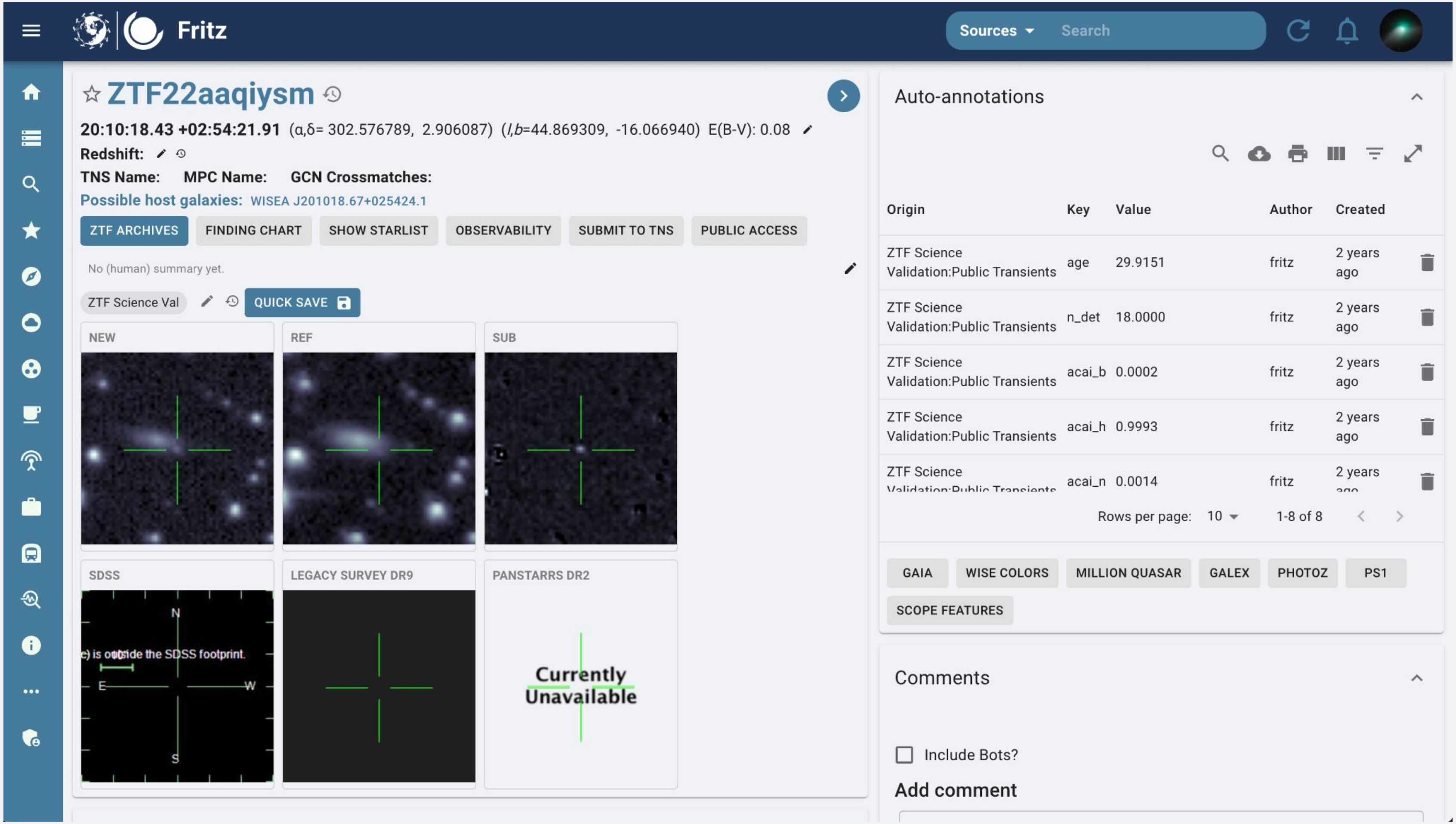
```
{
  "catalog_name": "ZTF_alerts",
  "filter": null,
  "limit": 1,
  "max_time_ms": 60000,
  "object_coordinates": {
    "test": [
      30, 45
    ]
  },
  "projection": null,
  "radius": 30,
  "skip": null,
  "sort": null,
  "unit": "Degrees"
}
```

21ms 2.12 kB 200 OK

application/json

```
{
  "status": "success",
  "message": "Cone Search on ZTF_alerts completed",
  "data": {
    "test": [
      {
        "_id": 3081488516015010000,
        "objectId": "ZTF18aabgyze",
        "candidate": {
          "jd": 2460835.9885185,
          "band": "r",
          "pid": 3081488516015,
          "diffmaglim": 18.469295501708984,
          "programpi": "Prince",
          "programid": 1,
          "candid": 3081488516015010000,
          "isdiffpos": false,
          "nid": 3081,
          "rcid": 60,
          "field": 1695,
          "ra": 26.0336366,
          "dec": 40.2225264,
          "magpsf": 16.93102264404297,
          "sigmapsf": 0.046223316341638565,
          "chipsf": 1.7859874963760376,
          "magap": 16.85140037536621,
          "sigmagap": 0.09030000120401382,
          "distnr": 0.16483153402805328,
```

▶ BOOM



☆ ZTF22aaqiysm

20:10:18.43 +02:54:21.91 (α,δ= 302.576789, 2.906087) (*l*,*b*=44.869309, -16.066940) E(B-V): 0.08

Redshift:

TNS Name: MPC Name: GCN Crossmatches:

Possible host galaxies: WISEA J201018.67+025424.1

No (human) summary yet.

ZTF Science Val

e) is outside the SDSS footprint.

Auto-annotations

Origin	Key	Value	Author	Created
ZTF Science Validation:Public Transients	age	29.9151	fritz	2 years ago
ZTF Science Validation:Public Transients	n_det	18.0000	fritz	2 years ago
ZTF Science Validation:Public Transients	acai_b	0.0002	fritz	2 years ago
ZTF Science Validation:Public Transients	acai_h	0.9993	fritz	2 years ago
ZTF Science Validation:Public Transients	acai_n	0.0014	fritz	2 years ago

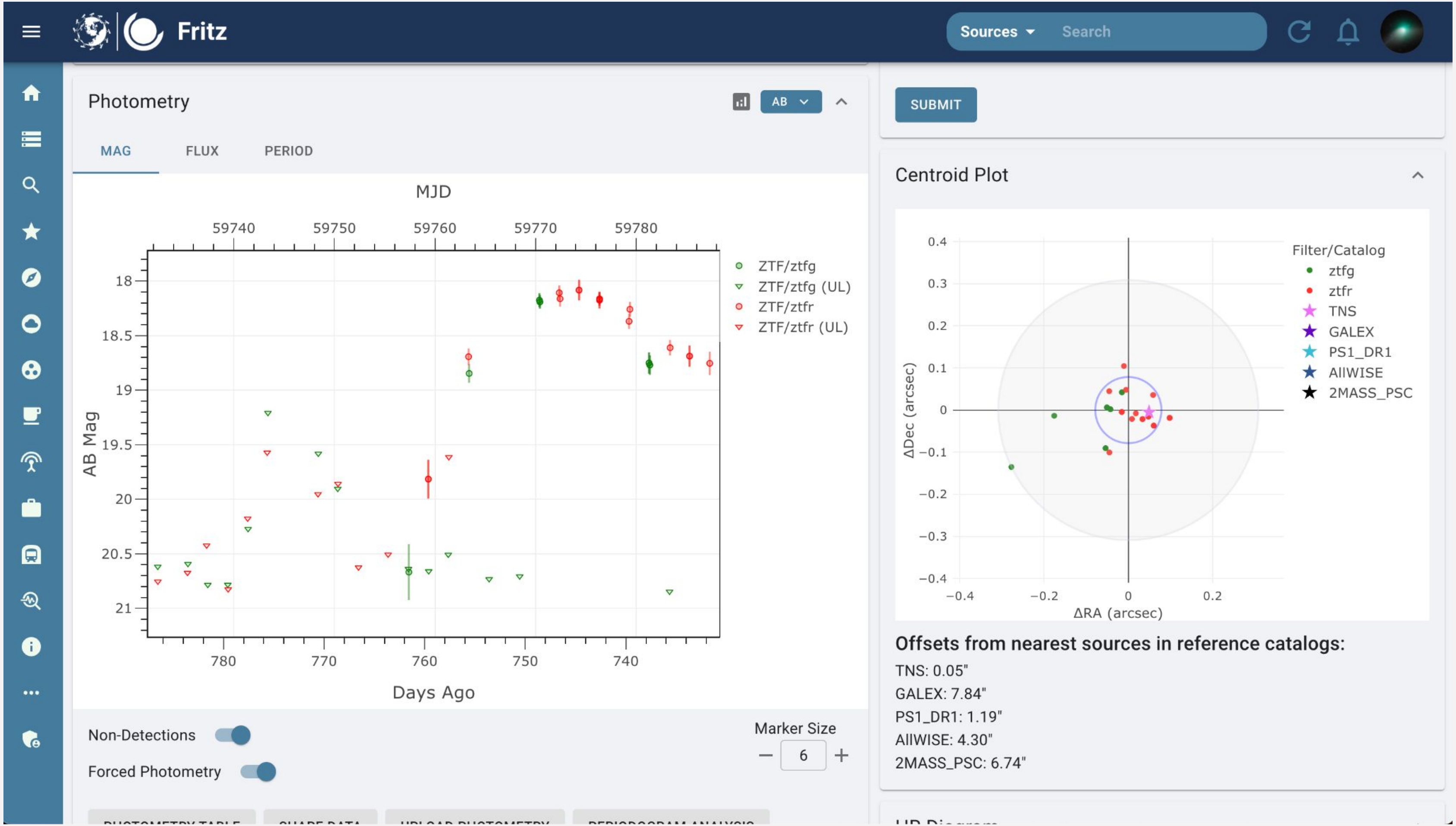
Rows per page: 10 1-8 of 8

Comments

□ Include Bots?

Add comment

BOOM



Offsets from nearest sources in reference catalogs:

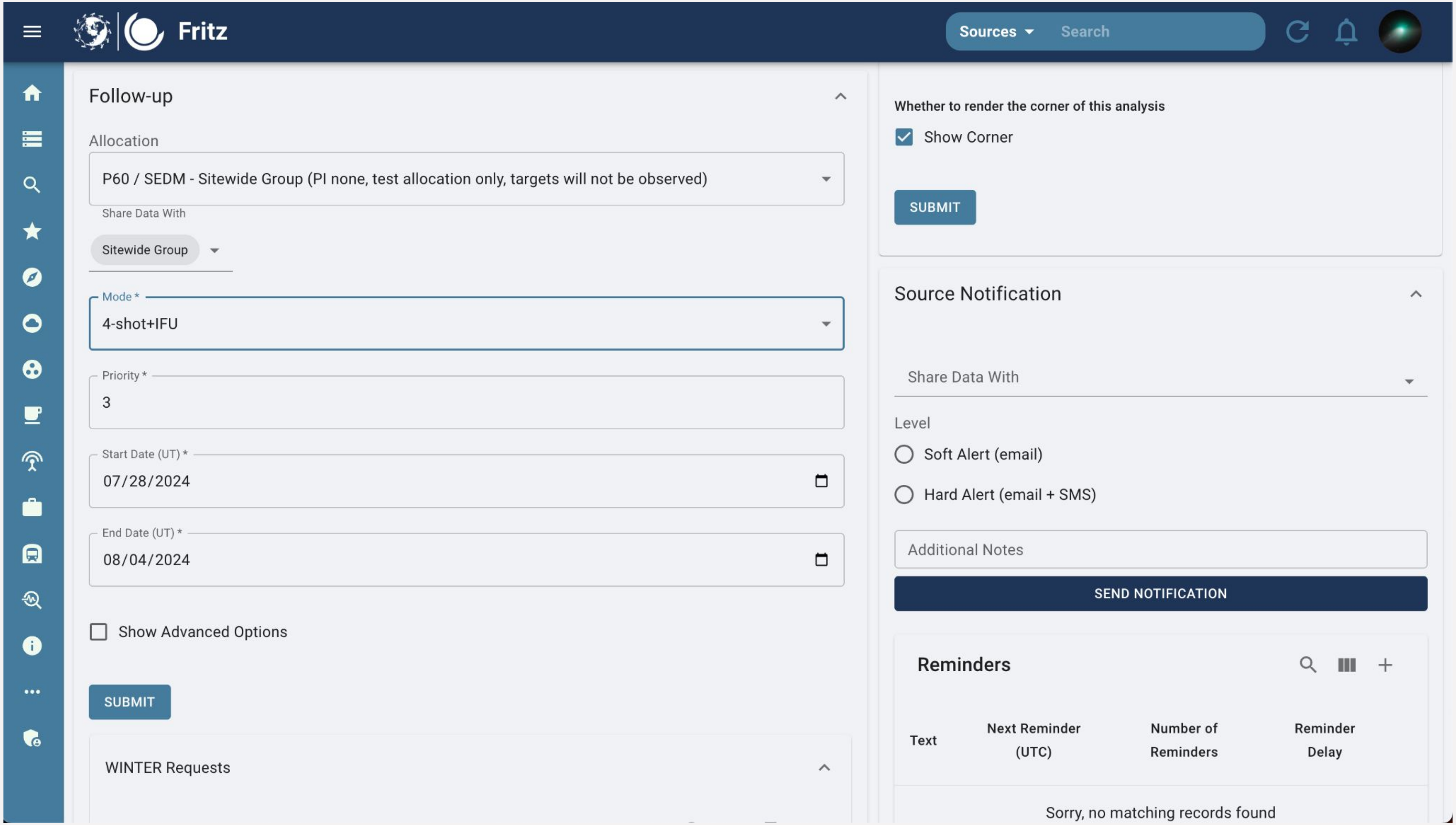
TNS: 0.05"

GALEX: 7.84"

PS1_DR1: 1.19"

AllWISE: 4.30"

2MASS_PSC: 6.74"



Follow-up

Allocation: P60 / SEDM - Sitewide Group (PI none, test allocation only, targets will not be observed)

Share Data With: Sitewide Group

Mode *: 4-shot+IFU

Priority *: 3

Start Date (UT) *: 07/28/2024

End Date (UT) *: 08/04/2024

☐ Show Advanced Options

WINTER Requests

Whether to render the corner of this analysis

☑ Show Corner

Source Notification

Share Data With

Level

○ Soft Alert (email)

○ Hard Alert (email + SMS)

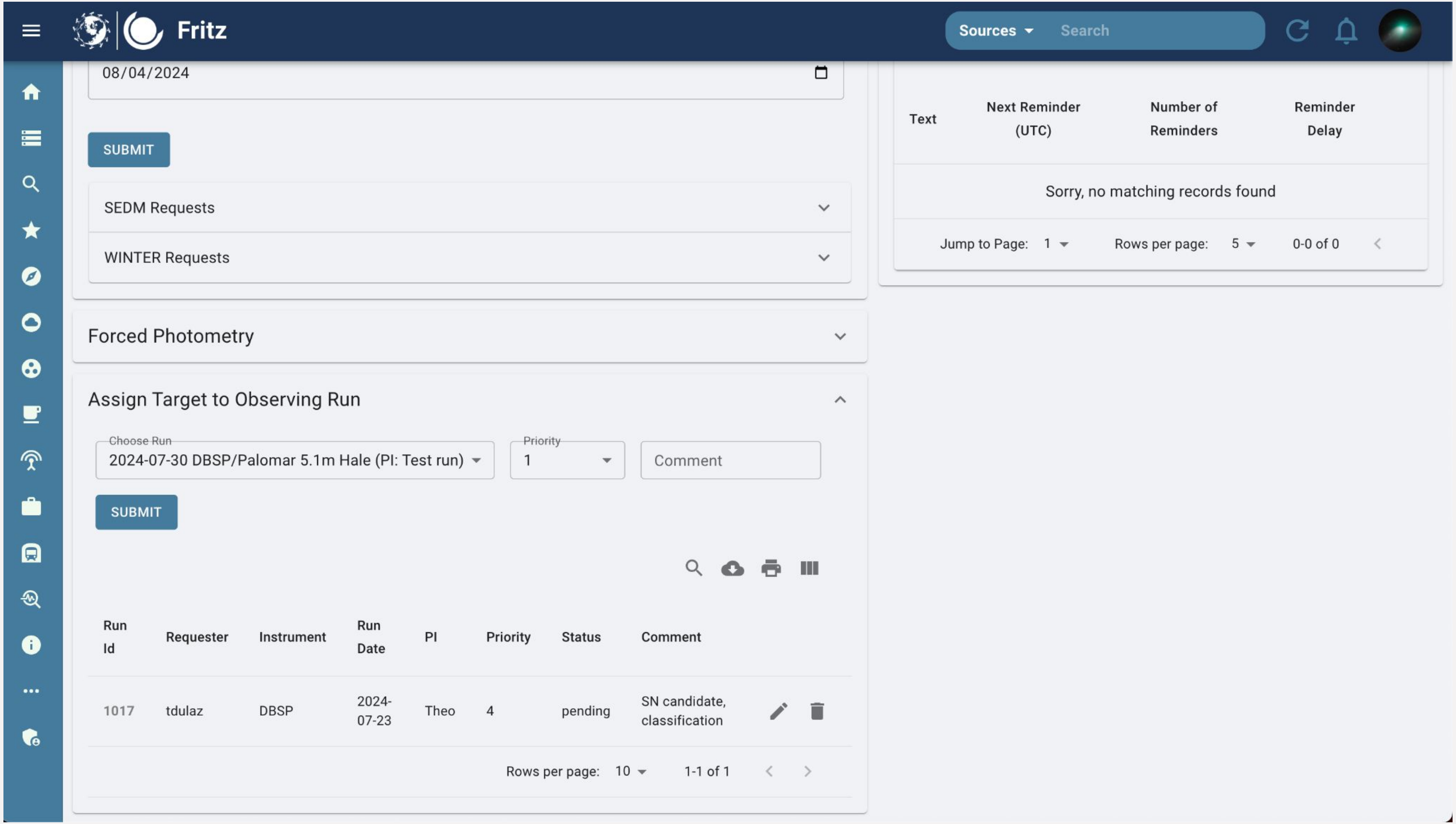
Additional Notes

Reminders

Text	Next Reminder (UTC)	Number of Reminders	Reminder Delay

Sorry, no matching records found

▶ BOOM



08/04/2024

Text	Next Reminder (UTC)	Number of Reminders	Reminder Delay

Sorry, no matching records found

Jump to Page: 1 Rows per page: 5 0-0 of 0

Assign Target to Observing Run

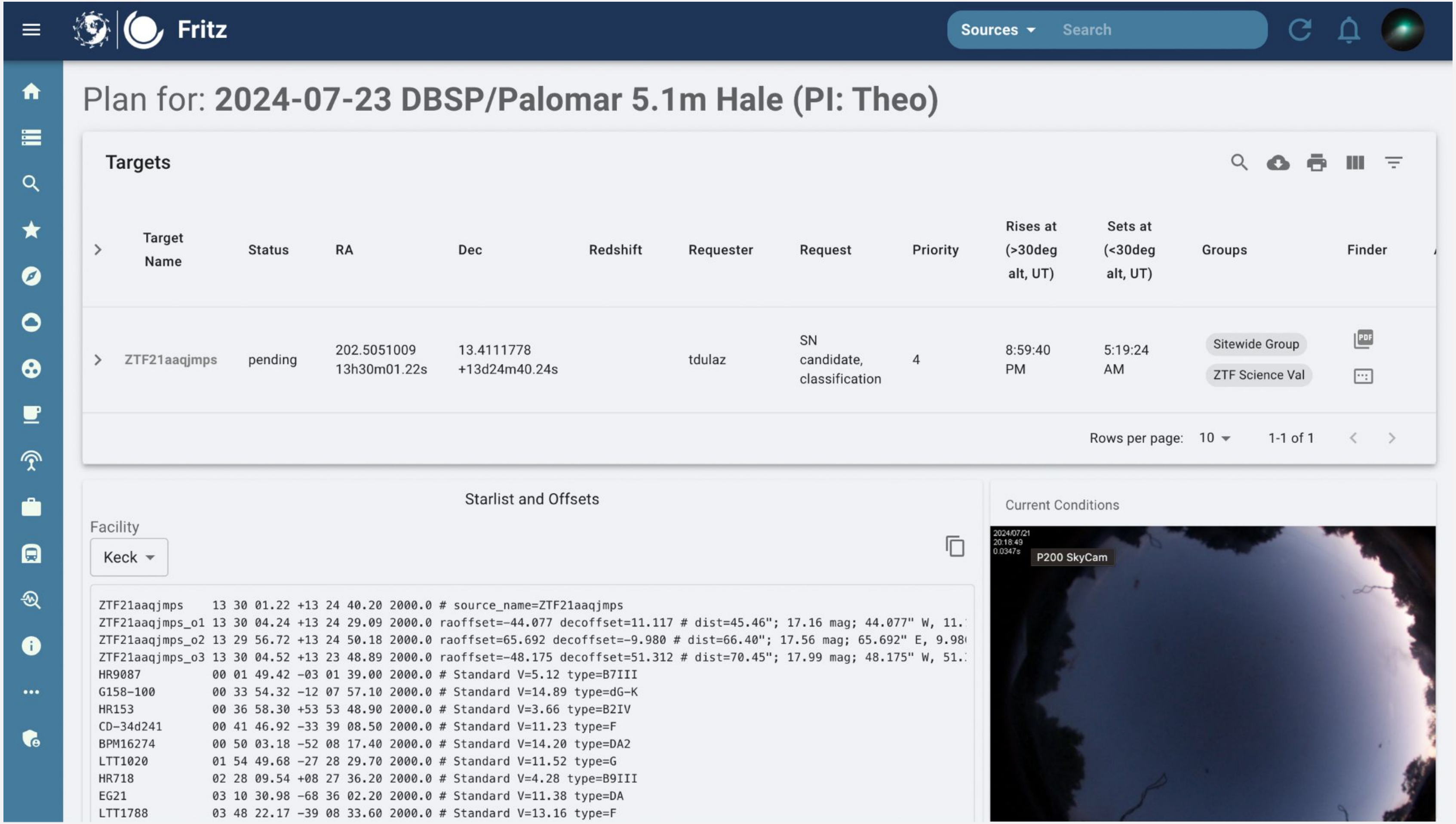
Choose Run: 2024-07-30 DBSP/Palomar 5.1m Hale (PI: Test run)

Priority: 1

Comment

Run Id	Requester	Instrument	Run Date	PI	Priority	Status	Comment
1017	tdulaz	DBSP	2024-07-23	Theo	4	pending	SN candidate, classification

Rows per page: 10 1-1 of 1



Plan for: 2024-07-23 DBSP/Palomar 5.1m Hale (PI: Theo)

Targets

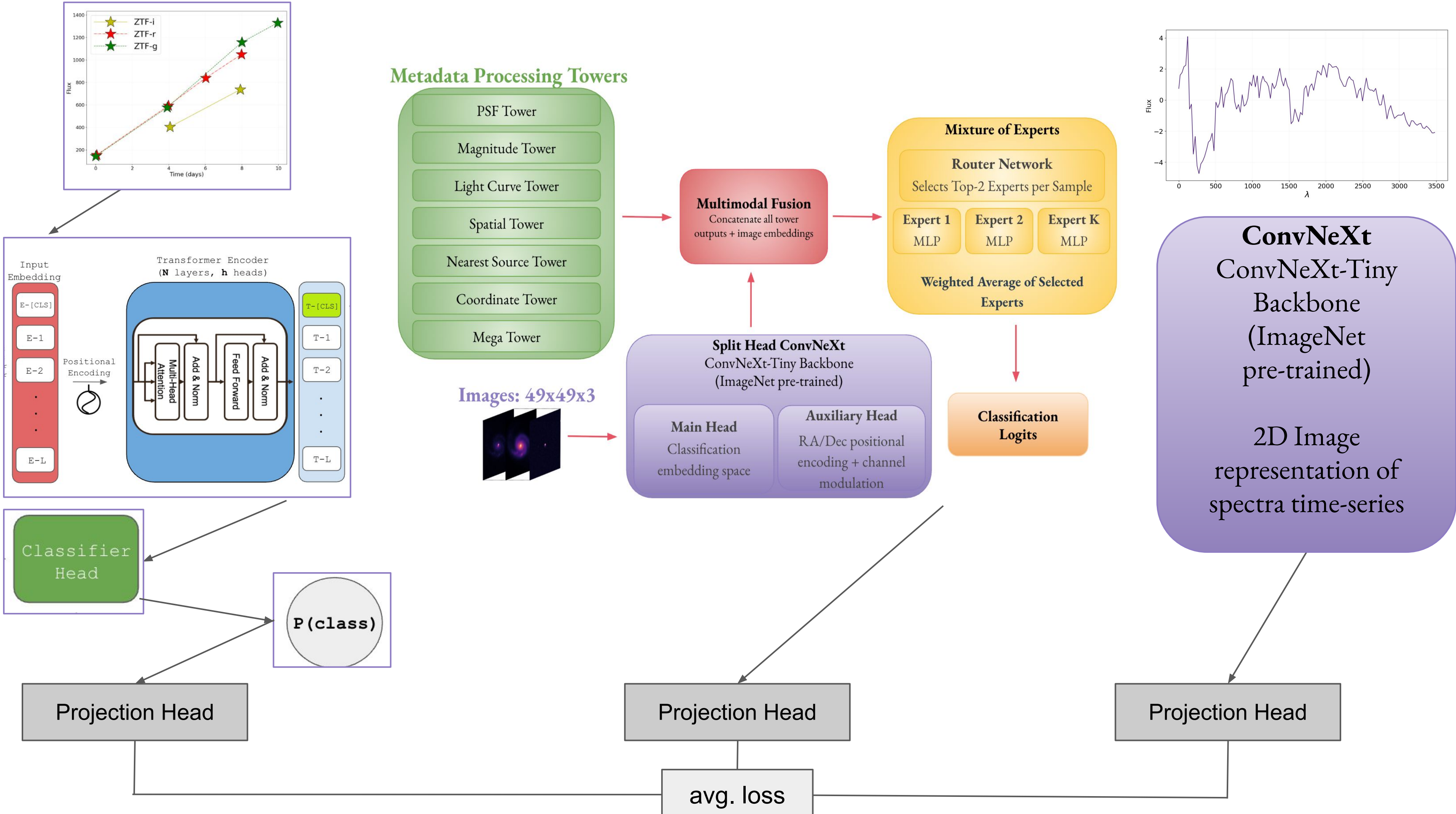
	Target Name	Status	RA	Dec	Redshift	Requester	Request	Priority	Rises at (>30deg alt, UT)	Sets at (<30deg alt, UT)	Groups	Finder
>	ZTF21aaqjmps	pending	202.5051009 13h30m01.22s	13.4111778 +13d24m40.24s		tdulaz	SN candidate, classification	4	8:59:40 PM	5:19:24 AM	Sitewide Group ZTF Science Val	

Rows per page: 10 1-1 of 1

Starlist and Offsets

Facility: Keck

```
ZTF21aaqjmps    13 30 01.22 +13 24 40.20 2000.0 # source_name=ZTF21aaqjmps
ZTF21aaqjmps_o1 13 30 04.24 +13 24 29.09 2000.0 raoffset=-44.077 decoffset=11.117 # dist=45.46"; 17.16 mag; 44.077" W, 11.
ZTF21aaqjmps_o2 13 29 56.72 +13 24 50.18 2000.0 raoffset=65.692 decoffset=-9.980 # dist=66.40"; 17.56 mag; 65.692" E, 9.98
ZTF21aaqjmps_o3 13 30 04.52 +13 23 48.89 2000.0 raoffset=-48.175 decoffset=51.312 # dist=70.45"; 17.99 mag; 48.175" W, 51.
HR9087          00 01 49.42 -03 01 39.00 2000.0 # Standard V=5.12 type=B7III
G158-100        00 33 54.32 -12 07 57.10 2000.0 # Standard V=14.89 type=dG-K
HR153           00 36 58.30 +53 53 48.90 2000.0 # Standard V=3.66 type=B2IV
CD-34d241       00 41 46.92 -33 39 08.50 2000.0 # Standard V=11.23 type=F
BPM16274        00 50 03.18 -52 08 17.40 2000.0 # Standard V=14.20 type=DA2
LTT1020         01 54 49.68 -27 28 29.70 2000.0 # Standard V=11.52 type=G
HR718           02 28 09.54 +08 27 36.20 2000.0 # Standard V=4.28 type=B9III
EG21            03 10 30.98 -68 36 02.20 2000.0 # Standard V=11.38 type=DA
LTT1788         03 48 22.17 -39 08 33.60 2000.0 # Standard V=13.16 type=F
```



Metadata Processing Towers

PSF Tower

Magnitude Tower

Light Curve Tower

Spatial Tower

Nearest Source Tower

Coordinate Tower

Mega Tower

Multimodal Fusion

Concatenate all tower outputs + image embeddings

Mixture of Experts

Router Network

Selects Top-2 Experts per Sample

Expert 1 MLP

Expert 2 MLP

Expert K MLP

Weighted Average of Selected Experts

Images: 49x49x3

Split Head ConvNeXt

ConvNeXt-Tiny Backbone (ImageNet pre-trained)

Main Head

Classification embedding space

Auxiliary Head

RA/Dec positional encoding + channel modulation

Classification Logits

ConvNeXt

ConvNeXt-Tiny Backbone (ImageNet pre-trained)

2D Image representation of spectra time-series

Input Embedding

Positional Encoding

Transformer Encoder (**N** layers, **h** heads)

Classifier Head

P(class)

Projection Head

Projection Head

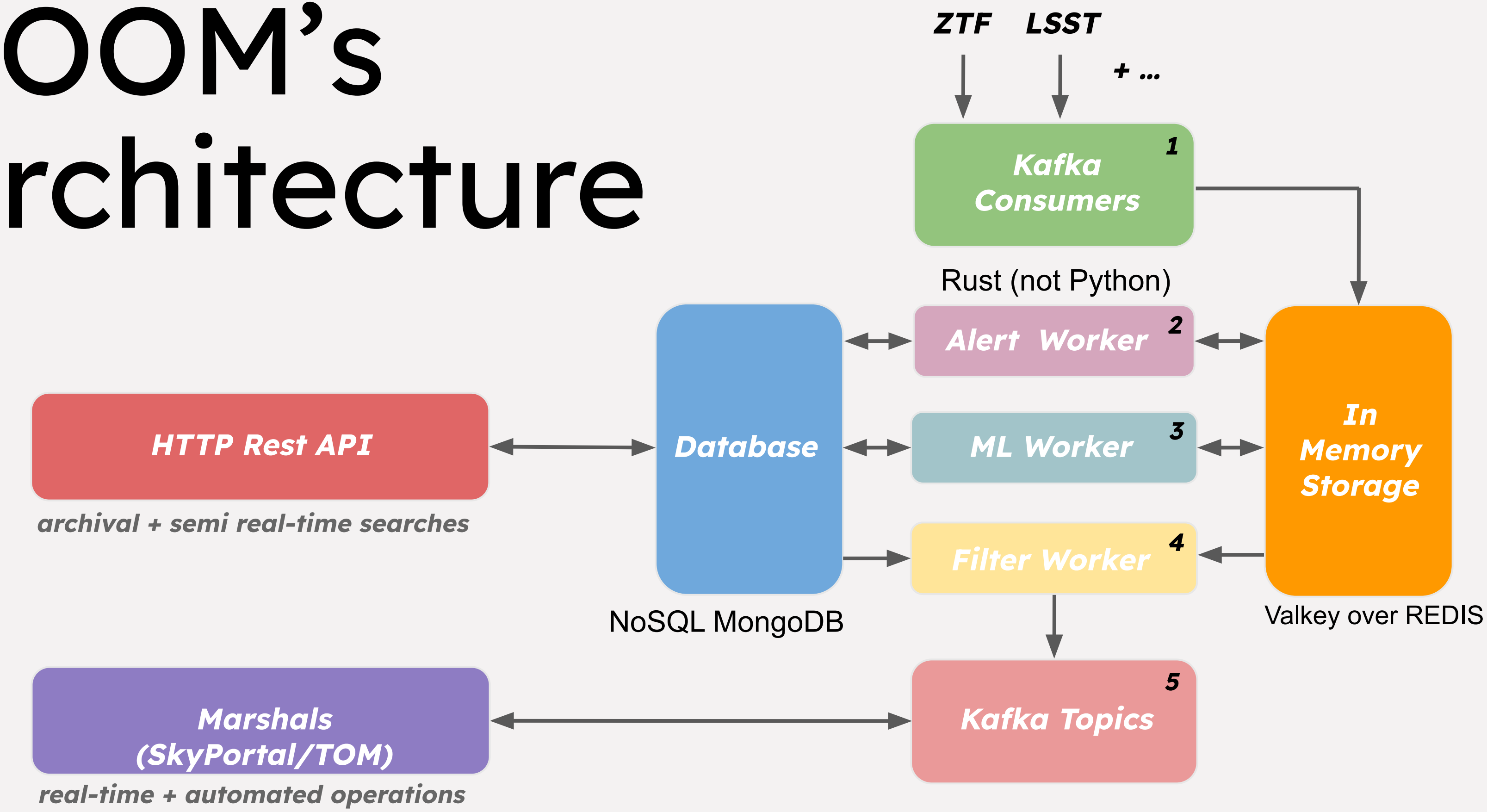
Projection Head

avg. loss

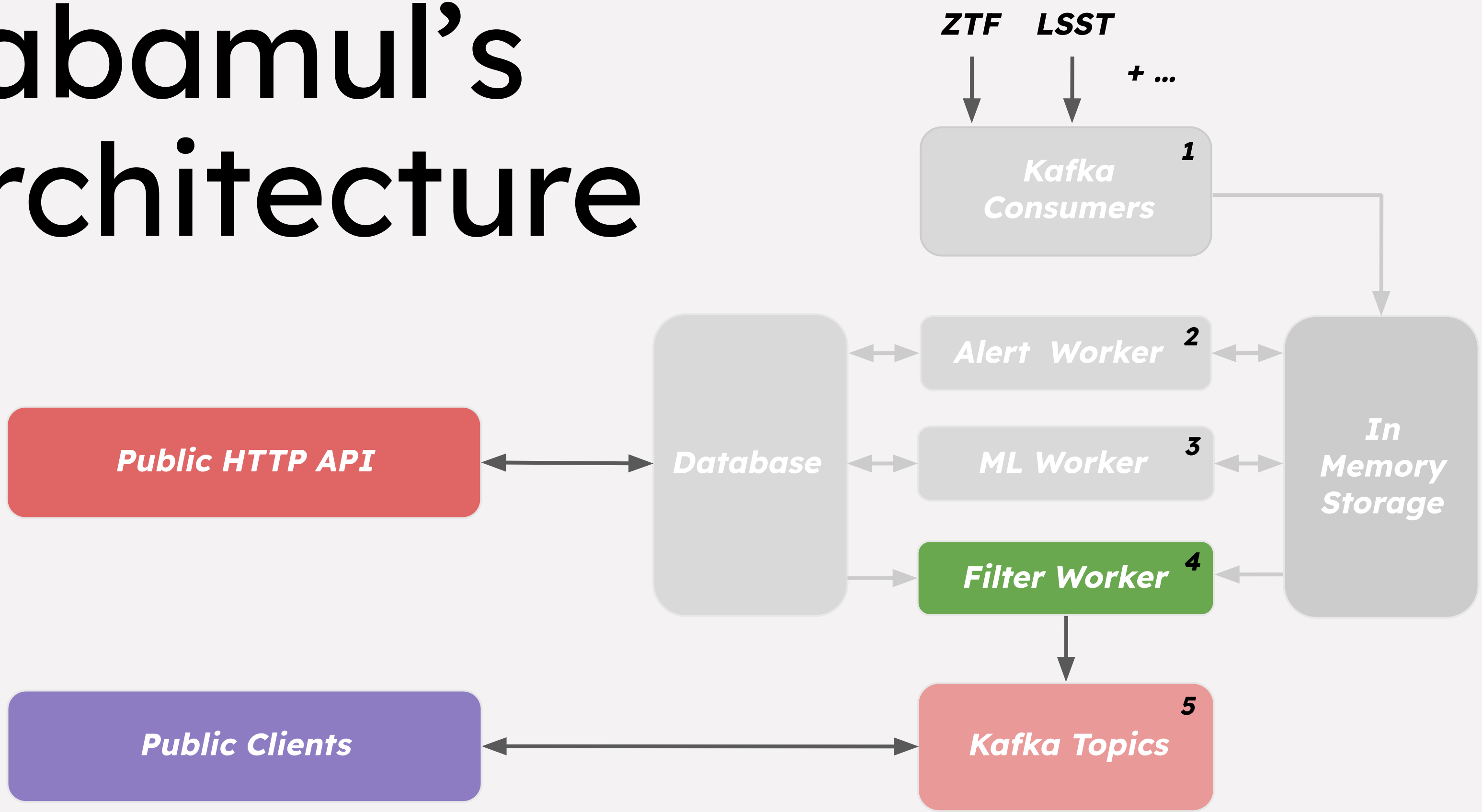
A solution: BOOM

- **Performance:** Rust-based, lowering the memory & CPU usage and increasing speed = more parallelization possible + less latency
- **Modularity:** Instead of one big processing step that does everything, we want a cluster-like architecture with specialized workers/processes. This way, we can scale up & down each processing step individually, based on the time they need to run.
- **Distributed:** Ideally, we want a distributed system where we can have multiple machines join the cluster to scale vertically

BOOM's Architecture

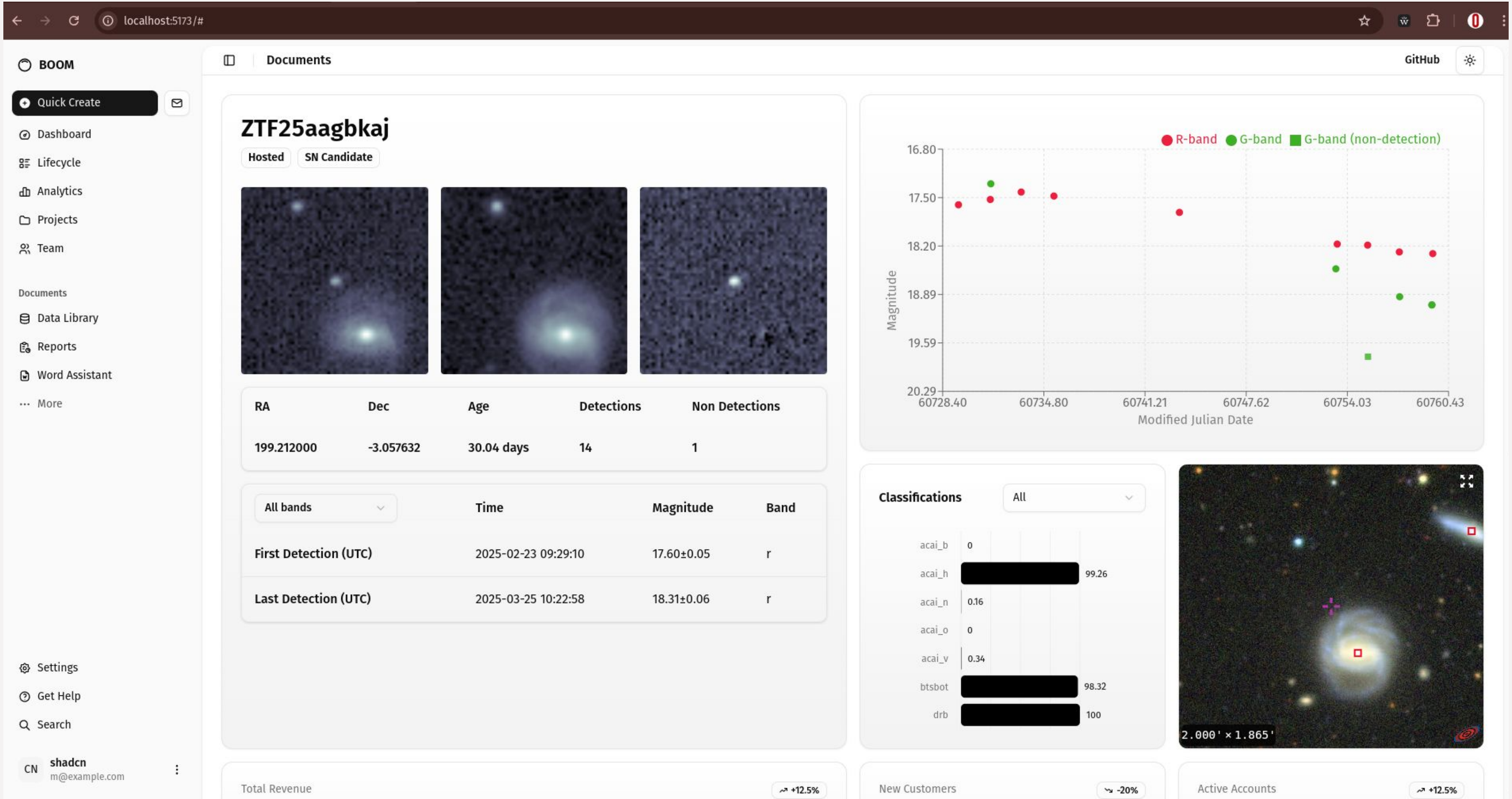


babamul's Architecture



- **A Frontend?**
  - What we’ve learned from deploying a marshal for ZTF is that **frontend is king**, over ⅔ of the users rely only on the web app. Similarly, it is pretty clear that amongst the existing brokers, those with the best feature rich vs complexity ratio are the most popular. **A front-end puts you on the map**, and becomes the **entry point** for users of all technical levels.
  - A fantastic feature would be to let users (optionally) create an account on that platform, that they can link to their marshal(s) account(s), to have access to extra features (e.g. see spectra or additional photometry points, if available, see existing classifications if any, …). This makes the broker’s frontend a lot more useful, while encouraging users to deploy or get access to a marshal to benefit from these.

BOOM



ZTF25aagbkaj

Hosted SN Candidate

RA	Dec	Age	Detections	Non Detections
199.212000	-3.057632	30.04 days	14	1

All bands	Time	Magnitude	Band
First Detection (UTC)	2025-02-23 09:29:10	17.60±0.05	r
Last Detection (UTC)	2025-03-25 10:22:58	18.31±0.06	r

Classifications All

acai_b	0
acai_h	99.26
acai_n	0.16
acai_o	0
acai_v	0.34
btsbot	98.32
drb	100

Total Revenue ↗ +12.5%

New Customers ↘ -20%

Active Accounts ↗ +12.5%