

AMPEL v 0.4



AMPEL stands for Alert Management, Photometry and Evaluation of Lightcurves. AMPEL is a partnership and public alert broker and manager and processes all ZTF alerts and provides a frame-work for both automatic real-time action and comprehensive full sample transient analysis.

(Full system overview can be found in previous collaboration presentations.)

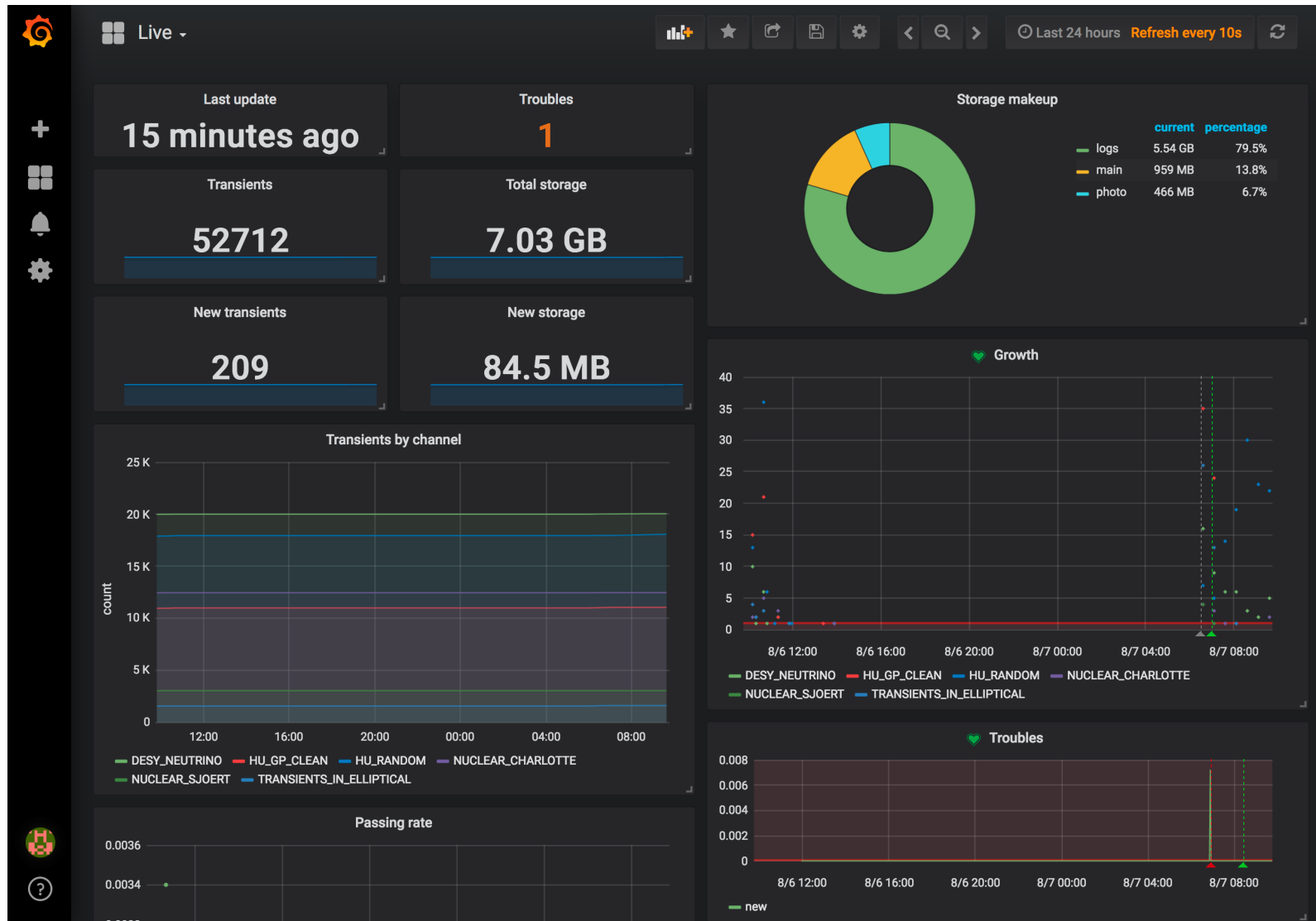


Current status

- Ampel online since June
- Multiple partnership channels
 - IceCube all-sky and ToO
 - ZTFnuclear core and off-core
 - Cosmology: HU clean + 2x lensing
- First public channels taken into production

Development by large group of people:
Valery Brinnel, Jakob van Santen, Matteo Giomi, Robert Stein, Sjoert van Velzen, Ludwig Rauch, Mickael Rigault, Maayane Soumagnac ...

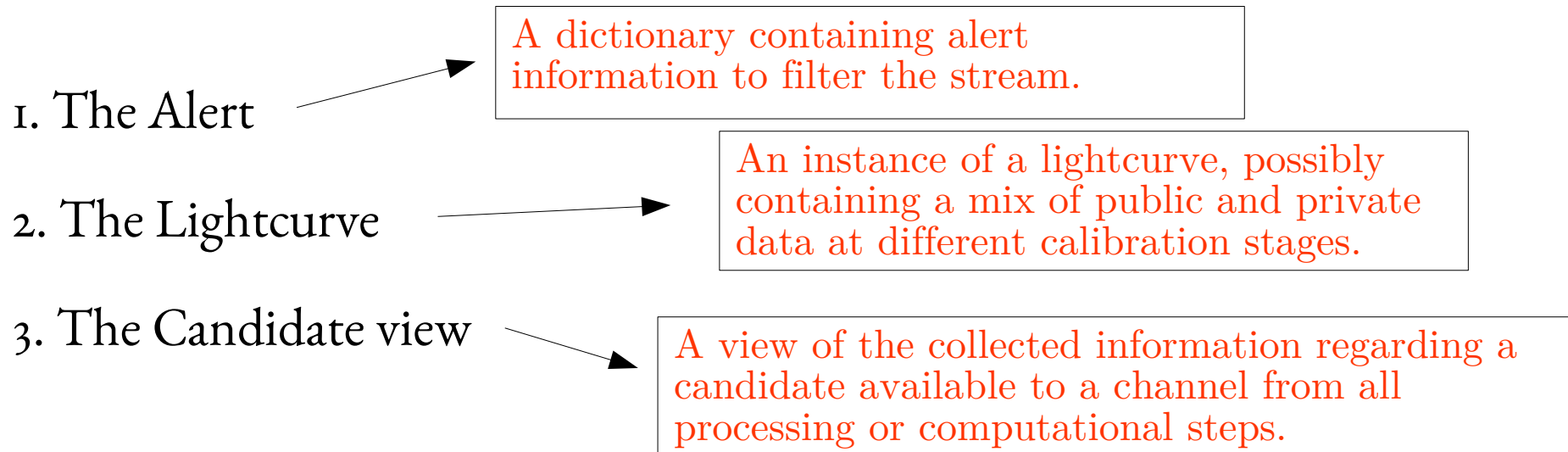
More dashboards



When reprocessing the full archive, we currently do 1000 alerts/s (real-time rate never this high).

Update 1: Interface Objects

A science program can be described through how it interacts at different information levels. In Ampel this means contributed code works through implementing abstract base classes which is provided one type of object:



At each stage you have access to *resources* such as catalog matching.

Ampel ensures that alert processing is done on all alerts, lightcurve analysis run at all relevant photometry combinations and information provided to the candidate view to facilitate a reaction.

Update 2: Container build & release plan

Developed routines for combining Ampel core code with user contributions*, add-on resources and the alert processor into containers.

The AMPEL server always runs one of these. Replacing the active version will be called a *release*, and typically happens on monthly time-scales.** *Between these we do not modify the online Ampel instance!*

* Users contributions are provided as github repositories where a given version is incorporated into the release.

** A new release does not mean that the active transient DB is reset.

Update 3: Target of Opportunity

A *ToO channel* continuously matches an (initially empty) database of on-sky areas. A trigger adds an entry to this live DB. Events will be reprocessed from a prior start date (to look for pre-existing transients) and new alerts matched until an end date.

Sample ToO trigger:

ToO_channel_name RA DEC radius start_time end_time

As for any channel, appropriate actions can be defined for matches (e.g. slack notice).

Example: The HU Clean Channel

Aim: A robust SN/AGN channel based on public data. A lightcurve fit can be used to find SNe close to peak.

Specified:

- 1) Use the *DecentFilter* with parameters 2+ positive detections, loose cuts on RB and image quality, no known PS star, no known Gaia proper motion.
- 2) Use *SNCOSMO* to fit a lightcurve with parameter specifying the SALT2 template.
- 3) Use the Slack and Marshal publisher interfaces with parameter specifying what is transferred to Slack and when these are run (once daily).

(We help with the configuration (straightforward), hard part is to define your science goals.)

Example: The HU Clean Channel

Final result provide 0.6% acceptance rate, roughly 100 transients/night.

Accepts (i) SN, (ii) core activity (iii) stars at the detection limit (easy to distinguish).

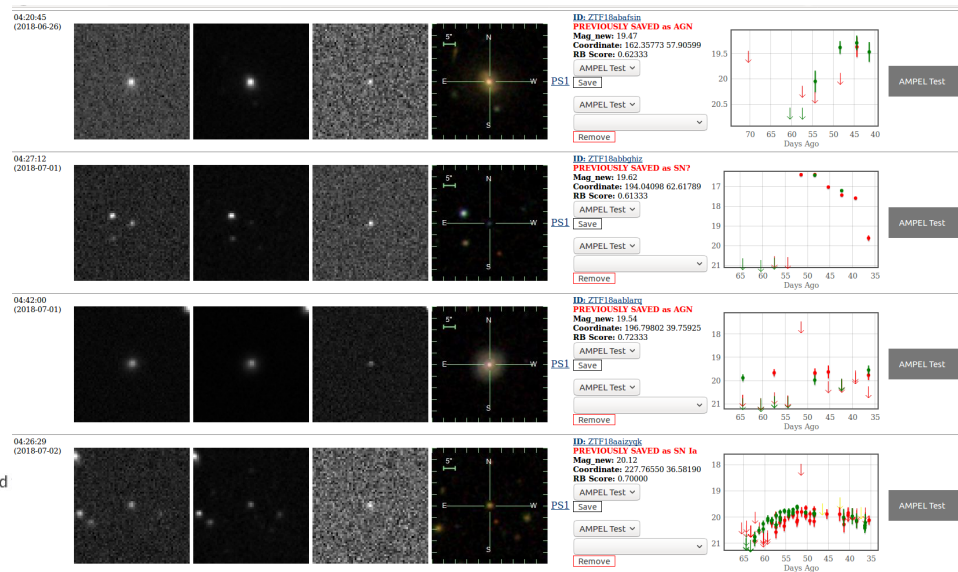
- # whitepaper
- # ztf_cosmos
- # ztf_lensing
- Direct Messages
- slackbot
- Jakob Nordin (you)
- Ariel Goobar
- Kerstin Paech
- Mickael Rigault
- Mickael Rigault, Ulrich ...
- Rahman Amanullah
- Rahman Amanullah, Se...
- Raphael Ferretti
- Seméli Papadogiannakis

AMPEL-live APP 11:45 AM
UPDATE! Alert summary for 2018-08-06.
WOW!!! We found 612 transients last night. 🎉 That's loads! Now we just need
The results are summarised below.

AMPEL-live APP 11:45 AM
Summary: 2018-08-06 ▾

```
1 , ztf_name, ra, dec, magpsf, sgscore1, rb, most_recent_detection, first_detection, n_detections, distnr, distpsnr1, isdiffpos, _id, OKC_LENSSED, TRANSIENTS_IN_ELLIPTICAL
2 0, ZTF18aaqduja, 200.1781001, 60.2571614, 18.5066032409668, 0.354059994220734, 0.509999990463257, 2458336.703287, 2458258.6808565, 53, 0.202585875988007, 0.175882413983345, True, !
3 0, ZTF18abjtgdo, 226.0983719, 61.6214602, 18.96364402770996, 0.05900000035762787, 0.7066666483879089, 2458336.7387153, 2458326.6770023, 32, 1.3529012203216553, 1.700149655342102,
4 0, ZTF18aazvmeg, 323.8785656, 45.0041293, 15.2517652511597, 0.5, 0.33666667342186, 2458336.847338, 2458280.9281944, 75, 0.37709578871727, 0.51721978187561, True, 53747564041501507:
5 0, ZTF18abkijyz, 254.9458714, 23.6082522, 20.026100158691406, , 0.2933329939842224, 2458336.7196412, 2458324.7110764, 20, 7.468699932098389, , t, 576315420815015010, False, False
```

 Full Photometry: 2018-08-06
21 MB CSV



Q: How can I log into AMPPEL and look
at my candidates?
(Alt. Where is the web query form?)

A: You cannot, both by design and necessity. Let me explain...

What do we want from a broker?

(and what is one?)

A software/hardware product taking care of transients. Could focus on many things:

1. Speed (eg Antares)
2. Interactivity (eg the Growth marshal)

What do we want from a broker?

(and what is one?)

A software/hardware product taking care of transients. Could focus on many things:

1. Speed (eg Antares)
2. Interactivity (eg the Growth marshal)
3. Continuity – keep track of evolving photometry & software
4. Flexibility – adoptable to specific science cases
5. Reproducibility – what implicit/explicit selection was made?

Prioritizing among these goals create different products.

Ampel: priorities and chain of consequences

Most interesting priority for future transient astronomy looks (to me) like:

1. Reproducibility
2. Flexibility
3. Continuity
4. Speed
5. Interactivity

Ampel: priorities and chain of consequences

Most interesting priority for future transient astronomy looks (to me) like:

1. Reproducibility

Frozen software (“containers”) and alert archives, virtual runs, minimal manual interaction.

2. Flexibility

Modular setup, python, contributed code, database.

3. Continuity

Keep track of changing photometry and typing, option to run new/old software with new/old photometry.

4. Speed

Dedicated machine @ CS, slow methods run on few candidates or offline, parallel instances tuned for science case

5. Interactivity

No direct access (container frozen and public cannot access CS). Interact w. offline DB copies or through exported content.

Where we hope to be in 6m

- AMPEL intro paper
- Interactive tool to *create* channels
- Full development version to test contribution code
- Computation (T₂) modules: photo-z, lightcurve fit (multiple z-option), AGN finder, photometric typing (ML and non-ML)
- Reaction (T₃) modules including telescope ranking/finding charts and TNS/Marshal *interaction*
- Guidelines for rerun of archive alerts based on container versions.
- Candidate purge.

We we can do this week

If you have ideas for what could/should be done, multiple people here can help flesh that out in Ampel lingo.

With Ampel core completed we now focus on user tools and computational modules. Give us feedback regarding their priority!

Questions for discussion:

1. Alert archive, photometry retrieval
 - Currently we archive all alerts. Necessary?
 - Advanced modules will request photometry. Will forced photometry include public data?
 - Upper limits?
2. TNS submission.
3. Structuring spectroscopic typing
4. Robotic SEDmachine trigger