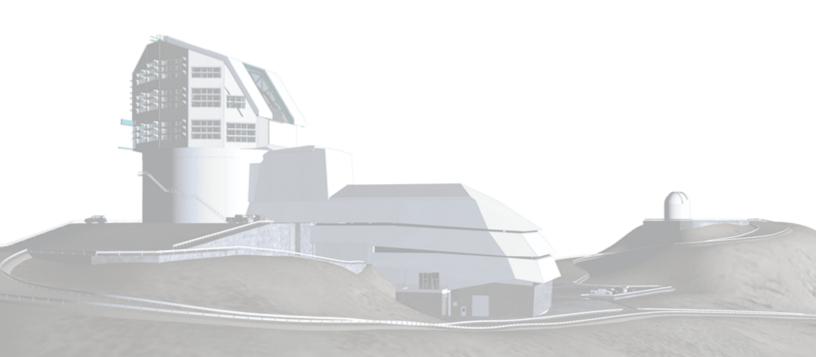## Vera C. Rubin Observatory
Data Management

Rubin/LSST Alert Filtering System

Eric C. Bellm, Leanne Guy

DMTN-226

Latest Revision: 2022-06-15

Abstract

This document presents a strategy for using the ANTARES alert broker as the Rubin Observatory Alert Filtering System.

Change Record

Version	Date	Description	Owner name
1	YYYY-MM-DD	Unreleased.	Eric Bellm

Document source location: `https://github.com/lsst-dm/dmtn-226`

Contents

Rubin/LSST Alert Filtering System

1 Background

During steady-state operations, Rubin Observatory's LSST will produce about ten million world-public alerts of transients, variable, and moving objects. Science users will work with community alert brokers to crossmatch, filter, and classify these alerts in order to identify the subset that require real-time followup observations. At the time the vision for this ambitious and comprehensive alert system was developed, no community alert brokers were functional. Accordingly, the Rubin project provided a fallback: a simple "alert filtering system" (AFS; §2) that the project would provide. The AFS would allow Rubin data-rights holders to apply simple user-defined or pre-defined filters to the Rubin alert stream. This minimized the risk that community alert brokers would not be available, or not provide the functionality required to fulfill users' needs.

Today, the situation is quite different. Seven community alert brokers[1] have been approved for direct access to the full Rubin alert stream, with two more planning to operate downstream of a full-stream broker. The brokers have mature systems and active user bases and are operating on live Zwicky Transient Facility alerts. Several brokers have functionality which mirrors that planned for the Rubin AFS. Accordingly, the rationale for a project-provided AFS is less clear.

However, it is reasonable for the Rubin project to ensure that Rubin data-rights holders have access to AFS-like capabilities throughout the survey, even if that service is provided by another entity. In this technote we propose a partnership with the ANTARES alert broker (Matheson et al., 2021), one of the seven approved full-stream brokers. Like the Rubin Operations project, ANTARES operates as part of NOIRLab.

In §2 we review the formal requirements for the Rubin AFS. §3 describes the ANTARES alert broker. In §4 we discuss steps necessary to allow ANTARES to supply AFS functionality to the Rubin data rights community.

---

[1] https://www.lsst.org/scientists/alert-brokers

2   Rubin Requirements for Alert Filtering

The top-level Science Requirements Document [LPM-17] describes the AFS in narrative language:

> The users will have an option of a query-like pre-filtering of this data stream in order to select likely candidates for specific transient type... Several pre-defined filters optimized for traditionally popular transients, such as supernovae and microlensed sources, will also be available, as well as the ability to add new pre-defined filters as the survey continues.

This is followed down into formal requirements as follows.

LSR, [LSE-29]:

LSR-REQ-0025 is supposed to be related but seems to be broken in current LSR

> Pre-defined filters optimized for traditionally popular transients shall be made available. It shall be possible for the project to add new pre-defined filters as the survey progresses.

LSR-REQ-0026

DMSR, [LSE-61]:

> A basic, limited capacity, alert filtering service shall be provided that can be given user defined filters to reduce the alert stream to manageable levels.

DMS-REQ-0342

> Users of the LSST Alert Filtering Service shall be able to use a predefined set of simple filters.

DMS-REQ-0348

> The LSST alert filtering service shall support `numBrokerUsers` (20) simultaneous users with each user allocated a bandwidth capable of receiving the equivalent of `numBrokerAlerts` (100) alerts per visit.

DMS-REQ-0343

Finally, the DPDD [LSE-163] provides a narrative description of the expected functionality from an end-user perspective.

> To directly serve the end-users, LSST will provide a basic, limited capacity, alert filtering service. This service will run at the LSST U.S. Archive Center (at NCSA). It will let astronomers create simple filters that limit what alerts are ultimately forwarded to them. These user defined filters will be possible to specify using an SQL-like declarative language, or short snippets of (likely Python) code. For example, here's what a filter may look like: *(Code snippet removed)*
>
> We emphasize that this LSST-provided capability will be limited, and is not intended to satisfy the wide variety of use cases that a full-fledged public Event Broker could. For example, we do not plan to provide any exclusive classification to a unique category of object. Following the SRD specification, however, we will provide a limited number of pre-defined filters for a small number of object types of common interest. These will answer non-exclusive questions such as "is the light curve consistent with an RR Lyra?", and will have potentially highly overlapping selections, designed to provide good completeness but perhaps only very modest purity. No information beyond what is contained in the VOEvent packet will be available to the pre-defined or user-defined filters (e.g., no cross-matches to other catalogs). The complexity and run time of user defined filters will be limited by available resources. Execution latency will not be guaranteed. The number of VOEvents transmitted to each user per visit will be limited as well (e.g., the equivalent of 20 full-size alert packets per visit per user, dynamically throttled depending on load). Finally, the total number of simultaneous subscribers is likely to be limited—in case of overwhelming interest, a TAC-like proposal process may be instituted.

Implicit in the discussion above is that the AFS should be broadly useful the widest potential range of alert-driven science. Additionally, consistent with the Rubin Data Policy [RDO-013], the AFS should be accessible to any Rubin Data Rights holder (subject to the capacity limitations described above).

While the alert data itself is world-public, access to Rubin-hosted services like the AFS is restricted to Data Rights holders. However, there is no restriction on third parties offering similar filtering services.

During Rubin Project development of the AFS, additional unofficial "desirements" were collected[2].

3 The ANTARES Alert Broker

ANTARES is a community alert broker that has been operating on ZTF alerts since 2018 and is approved for full stream access to LSST alerts. It provides a wide range of services to registered users from around the world, including alert visualization; rich positional and property searches; output Kafka streams; user watchlists; and rapid alerting via Slack. ANTARES also serves as the "upstream" broker for the SNAPS solar system broker.

ANTARES uses a "Locus" abstraction for timeseries data. New alerts are positionally associated with previous alerts at that location with a 1′′ tolerance. If no previous detections are present, a new Locus object is formed. Using a Locus allows users to make use of all of the historical lightcurve for an object rather than the limited window provided in each new alert. It also makes it convenient to attach multiwavelength crossmatch information.

ANTARES offers a capability for user filtering of the alert stream. Users may write filters in Python using the ANTARES development kit[3]. Filters may access all of the Locus information, including full timeseries history, catalog crossmatches, tags, and LIGO/Virgo detection information. Filters may provide outputs to the ANTARES database, Slack, and Kafka streams. These features provide valuable additional functionality beyond the minimal requirements of the Rubin AFS, which operate only on single alert packets with no crossmatching. ANTARES also provides access to predefined alert filters.

Filters are executed serially by "Pipeline Worker" jobs run inside Kubernetes, each of which attaches to a Kafka partition. Due to this design, user filters are not sandboxed from each other. Filters executed downstream can see tags applied by upstream filters and may be slowed by upstream filter execution. The ANTARES team mitigates these issues by reviewing filters before they are put into production and monitoring the operation of the filtering system. This increases the staff effort required to operate the ANTARES filtering system in steady state, but avoids the challenging problem of constructing a self-serve, fully sandboxed filtering system. Filter installation and updates thus cannot be conducted without staff intervention (i.e. within

---

[2]https://confluence.lsstcorp.org/display/~ebellm/Mini-Broker+Use+Cases+and+Draft+Requirements
[3]https://noao.gitlab.io/antares/filter-documentation/devkit/index.html

the night), although the ANTARES toolkit allows offline testing and tuning on sample data.

While having filters operate on Locus objects enables more scientific utility for stationary transients and variables, it limits the applicability of the ANTARES filtering system for Solar System Objects. Packet-based filtering of attributed Solar System Objects would be sufficient to satisfy the requirements of the Rubin AFS. This is not a current capability of ANTARES, although ANTARES does provide a stream of alerts from known Solar System Objects to the SNAPS solar system broker.

Currently, ANTARES executes 21 public and private filters on typically half a million alerts nightly. Filters in use today pass an average of 11% of alerts. (Since the filter outputs are implemented as tags, however, there are not strong technical pressures to limit the number of alerts which pass a filter, except in cases where the alerts are sent downstream through secondary Kafka streams.) These results suggest that ANTARES filtering can support the Rubin-required `numBrokerUsers` and `numBrokerAlerts`, although the ZTF-scale testing does not conclusively demonstrate performance at the scale of 10 million nightly Rubin alerts. However, the ANTARES team has performed scale testing that suggests acceptable performance at Rubin scale with planned hardware upgrades.

We thus conclude that it is highly likely that the ANTARES filtering service can meet the technical requirements of the Rubin AFS. The use of the Locus aggregation as an input to the filters provides scientifically valuable capabilities beyond the minimum requirements, but makes handling Solar System alerts more difficult.

4   Implementing the Rubin AFS with ANTARES

In January 2020, Rubin staff met with ANTARES personnel in Tucson to discuss whether the ANTARES filtering system could serve as the Rubin AFS, either as a software deliverable from the ANTARES team to be operated by the Rubin Project or as a service run by ANTARES. A follow-on video conference was held in April 2022. Participants in the latter teleconference agreed that the simplest and thus most cost-effective course of action would be to explore how the ANTARES-hosted filtering service could provide the functionality of the Rubin AFS.

We identify the following open questions which should be resolved to finalize this understanding:

- Can the ANTARES team confirm that their filtering service operates successfully at the throughput required for the Rubin AFS? Are there special considerations needed for ensuring access for Rubin Data Rights holders?

- How can we ensure that basic Solar System alert filtering can be conducted? Is additional ANTARES development needed? Could the SNAPS downstream broker team fulfill some or all of these needs?

- What level of additional operational effort (if any) relative to the baseline ANTARES plan is required to meet the Rubin needs? If so, can the appropriate color-of-money issues be resolved by Rubin Operations and the CSDC?

- What degree of interoperability between ANTARES and the Rubin Science Platform is desirable and achievable?

- What activities are needed to commission an ANTARES-hosted AFS, and on what timescale should they occur? (The relevant DM milestone DM-AP-14[4] is currently due in January 2024.)

- Are formal SLAs or MOAs between Rubin and ANTARES needed?

A   References

[RDO-013], Blum, R., the Rubin Operations Team, 2020, *Vera C. Rubin Observatory Data Policy*, RDO-013, URL `https://ls.st/RDO-013`

[LSE-29], Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2017, *LSST System Requirements (LSR)*, LSE-29, URL `https://ls.st/LSE-29`

[LSE-61], Dubois-Felsmann, G., Jenness, T., 2019, *Data Management System (DMS) Requirements*, LSE-61, URL `https://lse-61.lsst.io/`,
Vera C. Rubin Observatory

[LPM-17], Ivezić, Ž., The LSST Science Collaboration, 2018, *LSST Science Requirements Document*, LPM-17, URL `https://ls.st/LPM-17`

---

[4]`https://dmtn-158.lsst.io/#dm-ap-14`

[LSE-163], Jurić, M., Axelrod, T., Becker, A., et al., 2021, *Data Products Definition Document*, LSE-163, URL https://lse-163.lsst.io/, Vera C. Rubin Observatory

Matheson, T., Stubens, C., Wolf, N., et al., 2021, AJ, 161, 107 (arXiv:2011.12385), doi:10.3847/1538-3881/abd703, ADS Link

B   Acronyms