Rubin Observatory

Broker Selection & Hybrid Alerts

Eric Bellm
Alert Production Science Lead, Data Management

DMLT | Virtual | February 23-24, 2021

NSF AURA U.S. DEPARTMENT OF ENERGY Office of Science SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

The SAC will soon recommend a selection of community alert brokers.

Rubin Observatory

9 full proposals were submitted (out of 15 LOIs).
(All requested the full stream, few offered to forward.)

The committee will (at our request) select the top 5 (matching DMS-REQ-0391/numStreams = 5).

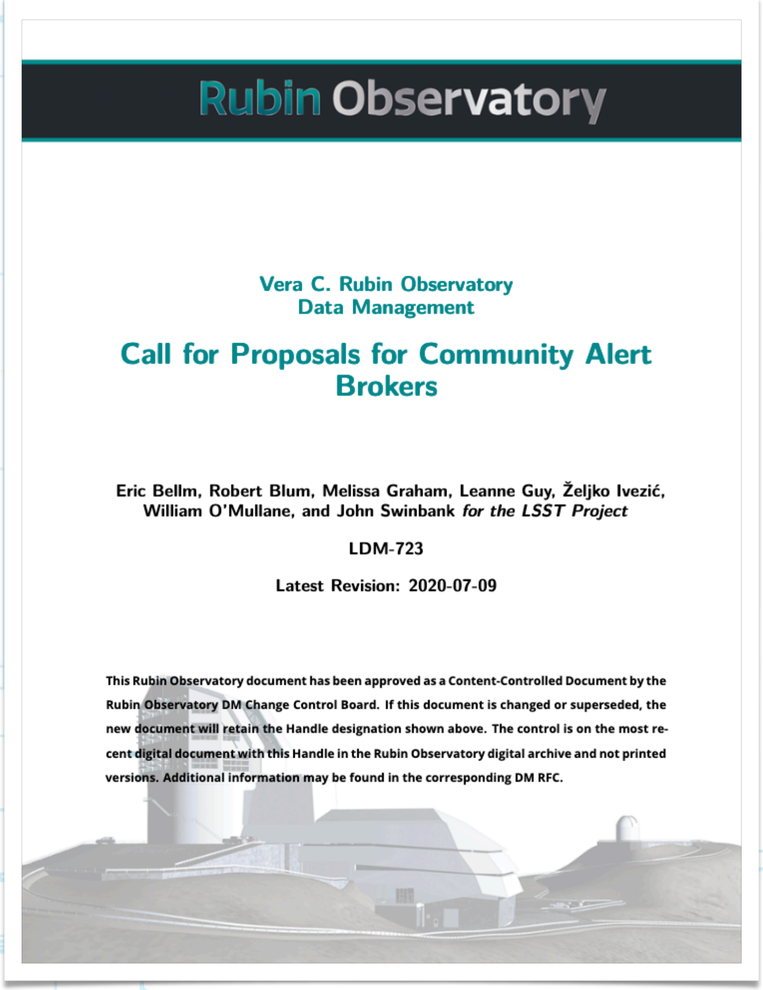
They also seem likely to strongly encourage support of at least two more brokers.

How do we respond to an N>5 recommendation?

- (next slide)

What are our next steps with the selected broker teams?

- MOUs?
- Preops, commissioning activities?


Rubin Observatory

Vera C. Rubin Observatory
Data Management

Call for Proposals for Community Alert Brokers

Eric Bellm, Robert Blum, Melissa Graham, Leanne Guy, Željko Ivezić, William O'Mullane, and John Swinbank *for the LSST Project*

LDM-723

Latest Revision: 2020-07-09

This Rubin Observatory document has been approved as a Content-Controlled Document by the Rubin Observatory DM Change Control Board. If this document is changed or superseded, the new document will retain the Handle designation shown above. The control is on the most recent digital document with this Handle in the Rubin Observatory digital archive and not printed versions. Additional information may be found in the corresponding DM RFC.

ls.st/LDM-723

We have a range of options for responding to a recommendation to support a larger number of full brokers.

Rubin Observatory

Number of full streams from the USDF is set by the combination of:

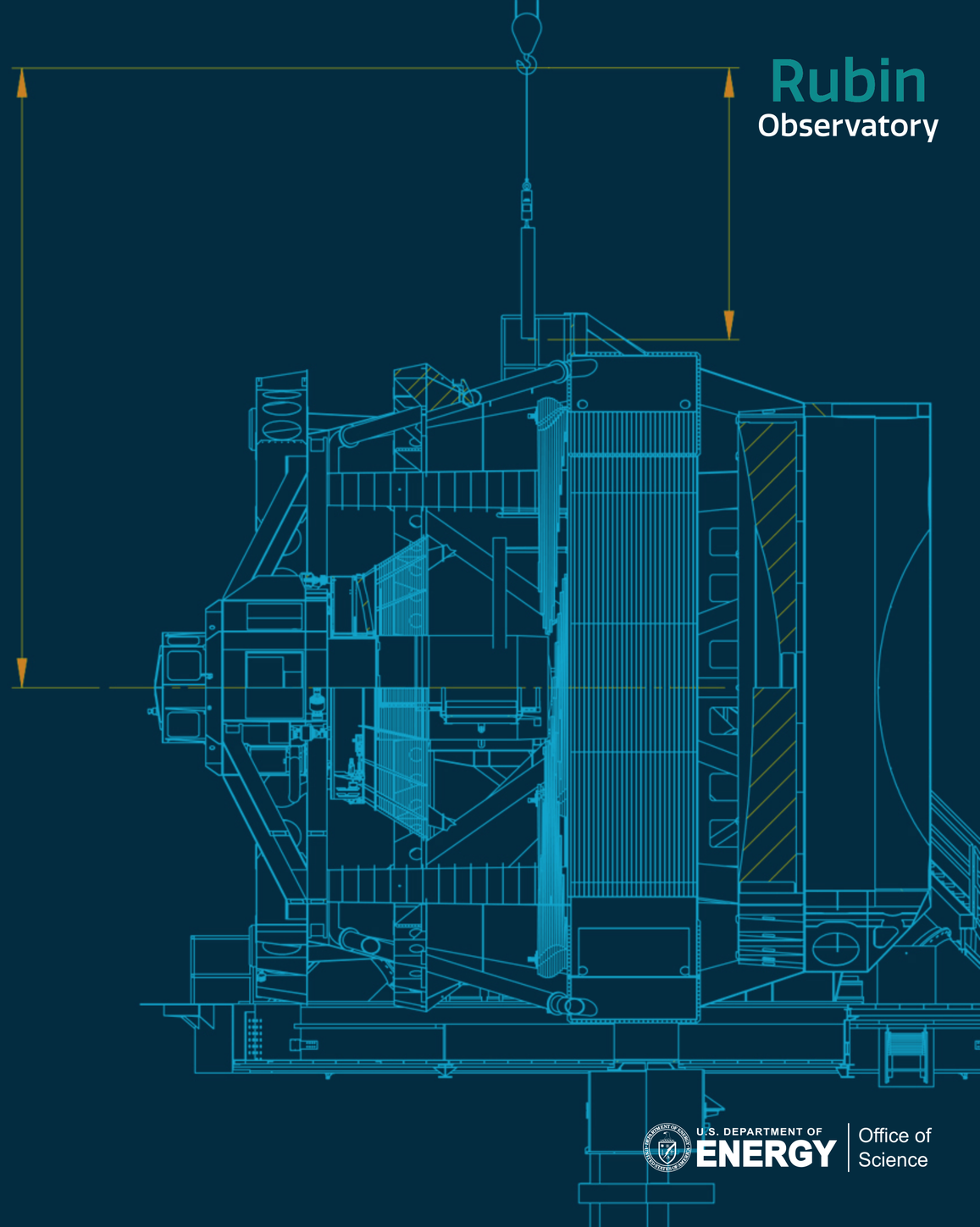
- available outbound bandwidth (NCSA used 10Gbps as a baseline)
- alert size
- allowed latency

USDF at SLAC may have more available bandwidth than 10Gbps baseline
⇒ if so, could support more brokers “for free”

Other options:

- only support N=5 brokers
- relax latency
- cut alert contents
- provide streams in the cloud & allow users to pay

Hybrid Alerts & the Alert Filtering Service



Rubin Observatory

U.S. DEPARTMENT OF ENERGY | Office of Science

The Alert Filtering Service

Rubin Observatory

We have requirements to produce an alert filtering service (AFS) and an alert database

- AFS must support 100 simultaneous users getting 20 alerts/visit
- filters act on single alert packet contents
- no latency requirement

It seems unlikely we'll be able to build the AFS by the end of construction.
AP demonstrated proof-of-concept filtering at required scale in the cloud, but a usable service requires much more.

Options previously discussed:

- descope AFS entirely (assume selected community brokers support that need)
- outsource AFS development

Today I want to explore a third option: using hybrid alerts as a replacement for the AFS

“Hybrid alerts” separate notification from data delivery.

Large alert packets contain all the key science information (and more) without followup queries to Rubin databases.

But they make bandwidth a bottleneck: users must go through a few community brokers or the AFS to get alert information, and those systems require careful engineering due to the data volume.

Individual science *users* will only ever want the full contents of a small subset of alerts.

Rather than filter full-sized alerts within the AFS, we can send a small notification for all alerts to science users, and allow them to retrieve the full-sized alert packets for a subset of alerts.

“Alerts on your laptop” could dramatically expand the number of users of the LSST alert stream, enhancing the creative use of the data increasing the science returns of the Project.

A minimal packet would allow users to identify a subset of relevant alerts.

Rubin Observatory

The “notification stream” (potentially still Kafka) would contain the *bare minimum* of science data to allow users to filter down 10x or more to alerts of interest

perhaps

- position, time, filter, observing program, magnitudes
- SSObject association, trailed source fit, extendedness, spuriousness score
- distance to nearest star, distance to nearest galaxy
- a handful of timeseries features
- a resource identifier linking to the full packet (or use the alertID & a resolver)

~200 *bytes* of data/packet! ⇒ allows ~thousands of users to connect

Complete alert packets would be served separately.

Rubin Observatory

Full alerts could be staged on an object store with an HTTP interface (for example)—this can be the required alert database

- easy to administer
- straightforward to set rate limits for individual users
- more performant (user ability to parallelize or schedule downloads)
- rate limits don’t have to be tied to individual visits

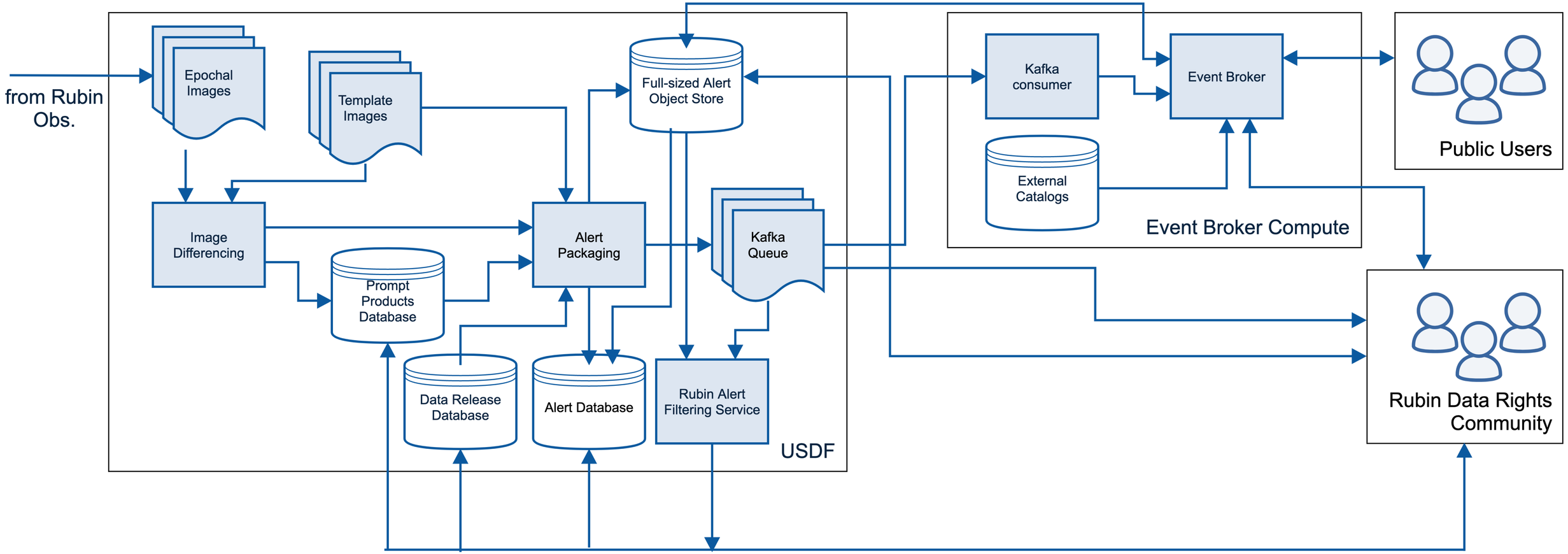
Having downselected to a few alerts from the notification stream, science users could then retrieve full alert data for more detailed analysis.

As an AFS replacement, direct access to the notification stream and full alerts would be restricted to data rights holders.

Public access to alerts would still be via community brokers.

In this architecture, data rights holders could choose to receive the thin—but complete—notification stream directly.

Rubin Observatory



Hybrid alerts would provide a better user experience than an on-project AFS and likely lower operations burden.

Rubin Observatory

On-Project AFS	Hybrid alerts
Small number of simultaneous users (~100)	Number of users receiving notification stream can be large—“permissionless”
User filters only operate on alert packet contents	Users can crossmatch to arbitrary catalogs, other streams, etc.
Users must write filters in Project-allowed language & runtime	Users are free to choose how to implement.
Project must monitor performance and security of user-submitted filters	Users are responsible for running their own filters and can use arbitrary computing resources.
Number of alerts sent is capped per visit	User can select how to allocate rate limit—e.g., retrieve all alerts from a single DDF.
Project must process multiple full-sized streams within the AFS	No on-project processing of alert stream contents.
Project must handle upload and validation of user filters and support user connection to filter output.	Project only supports user connection to notification stream.

Brokers could provide a similar service themselves, but there are no current live examples.

Rubin Observatory

A few brokers have also discussed providing “semantically compressed” streams of all alerts to end users, but none have done so to date.

A Project-provided service would replace the AFS and naturally leverage the Project’s existing commitment to build the Alert Database.

Users would be confident that a Project-provided service would be available in the long term.

Would full-stream brokers use the hybrid system as well?

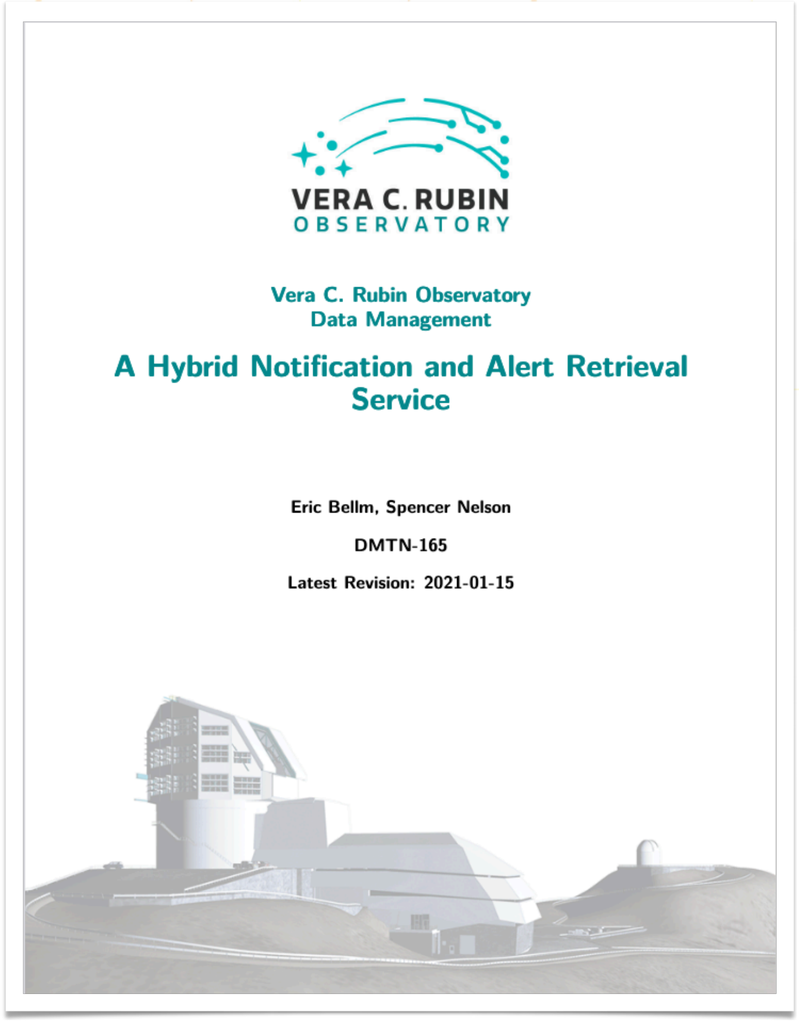
Rubin Observatory

With high enough rate limits and a performant alert DB, brokers could use the hybrid system instead of a full-sized alert stream to retrieve all alerts

- Requires an additional round trip for the brokers
- Still could meet formal OTT1 requirement for latency to brokers
- Provides easier management of Alert Distribution for Ops due to smaller packet sizes
- May provide technical advantages for the brokers due to the ability to parallelize or schedule downloads of full alerts
- Likely in practice to smooth demand for USDF outbound bandwidth
- Potentially a way to allow new or additional brokers to connect.

(Initial reactions from broker teams has been lukewarm.)

DMTN-165 discusses these issues further.



dmtn-165.lsst.io