



Dakota VOEvent Tools – Connecting to the VOEvent Network



Robert B. Denny, DC-3 Dreams SP
IVOA VOEvent Working Group

Abstract

The Dakota VOEvent Tools provide a freely available VOEvent backbone connectivity and distribution package for Linux, Mac OS X and Windows. Tools consist of a receiver/broker, a command line publisher, and a command line message validator.

With the Dakota VOEvent Tools, you can connect to the VOEvent network backbone and receive transient event alerts in real time. You can also publish event triggers to the backbone for others to receive. Finally, you can provide a broker service to the backbone, helping distribute messages from others around the world or from your facility All without writing any software.

Components

- **Publishing tool**
 - ✓ Broadcasts VOEvent to the net
 - ✓ Apply a digital signature to the message (optional)
- **Message Validator**
 - ✓ Checks VOEvent XML for validity
- **Receiver/Broker**
 - ✓ Listens for messages from remote broker
 - ✓ Validates incoming message against specification
 - ✓ Checks digital signatures on incoming messages
 - ✓ Delivers messages to pipeline (script or program)
 - ✓ Relays messages to other subscribers (broker)

What's Missing?

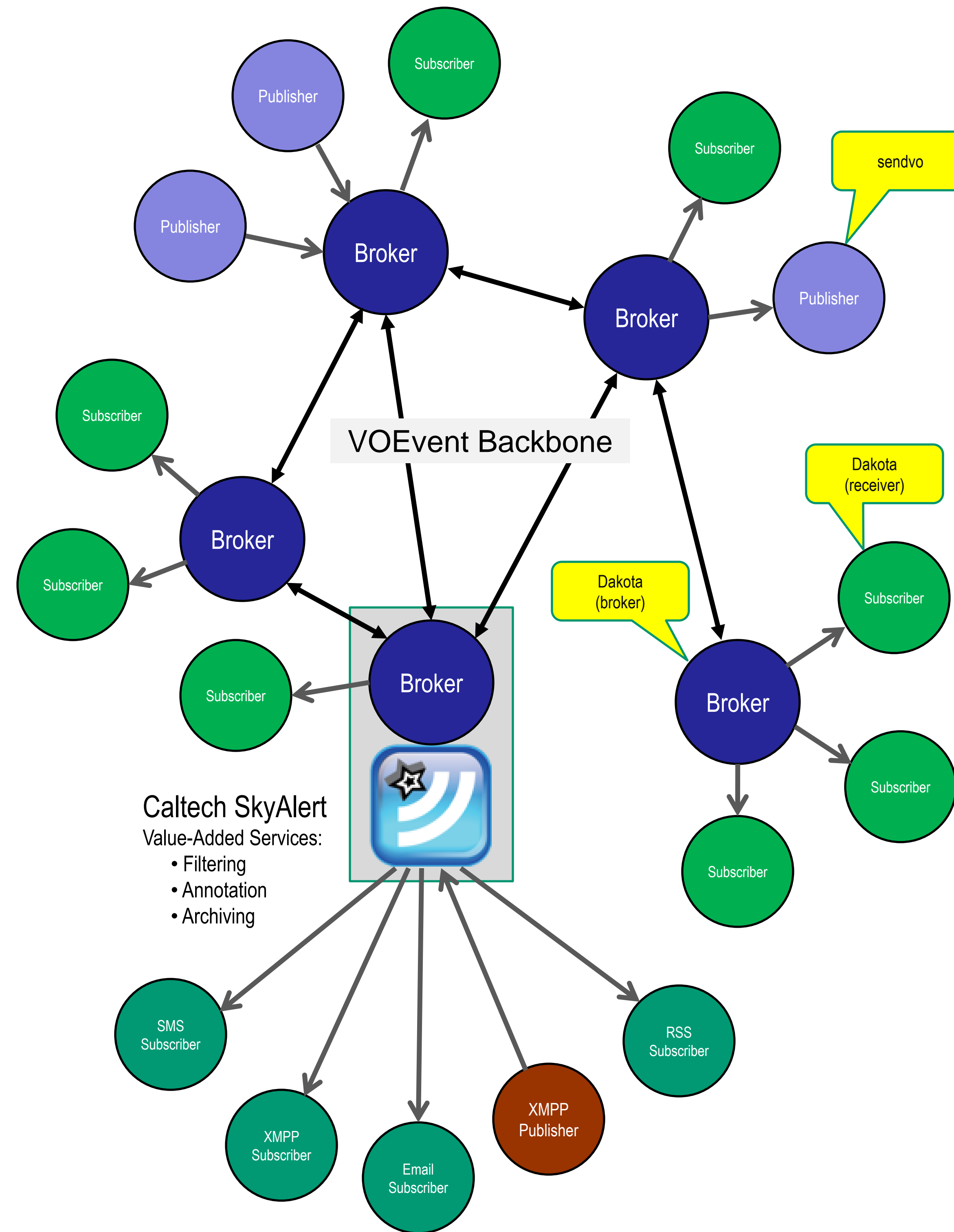
Tools for constructing and reading the VOEvent XML messages

The messages themselves are mission-specific, so it's nearly impossible to provide a "one-size fits all" set of tools for this.

The syntax and semantics of a VOEvent message are non-trivial. There are two problems to solve:

- Constructing a message from your data
- Extracting the data from a received message

Solutions range from manual angle-bracket picking and tag writing to machine-generated processors that extract data from messages, and turn data into messages. The latter technique requires substantially less effort while providing rigorous correctness of message structure and data encoding.



VOEvent Network Architecture

Technical Details

Dakota is a cross-platform open-source set of VOEvent tools which fully implement the VOEvent Transport 1.1 protocol for subscriber, publisher, and broker-to-broker operations. Tools include a receiver/broker, a message sender, and a message checker.

The receiver/broker can be operated as an end-point receiver without any message distribution or as a full broker. It has support for passing received messages to a script or program, allowing you to start a pipeline with incoming VOEvent messages.

The message sender **sendvo** is a command line tool that publishes a VOEvent message from a file or a standard-in stream to a specified broker. The broker's response is sent to standard out.

The message checker **checkvo** is a command line tool that performs a deep validation on a VOEvent message from a file or standard-in stream, and reports on its correctness.

The receiver/broker provides separate subscriber and publisher authentication via IP address white listing, as well as optional digital signature based authentication for publishers (only). In addition, it keeps a local copy of all VOEvent messages it receives (and optionally distributes), using this archive to prevent looping of messages received more than once.

Finally, the receiver/broker can be configured to start an external program or shell script on receipt of a message from a publisher or a remote broker. This allows you to immediately process received messages using the program or script of your choice. Thus, you have the flexibility of using Dakota as a receiver, then later enabling it to act as a broker while still serving as a receiver for your needs.

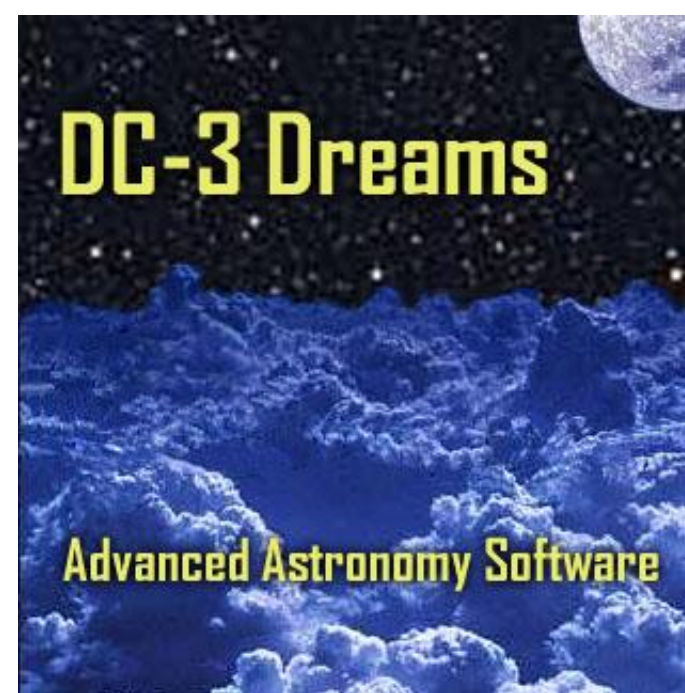
Taken together, the Dakota VOEvent Tools provide everything you need to publish and/or receive VOEvent messages, as well as providing a broker service for your installation or region.

Distribution and Support

The Dakota VOEvent Tools is an open-source project hosted on the SourceForge system. In addition, the author provides a web site that serves as a portal and basic information source. The author has committed to providing support and updates while a community is being formed.

The starting point is at

<http://voevent.dc3.com/>



A VOEvent Transient Astronomy Follow-Up Station

Robert B. Denny, DC-3 Dreams SP
 IVOA VOEvent Working Group



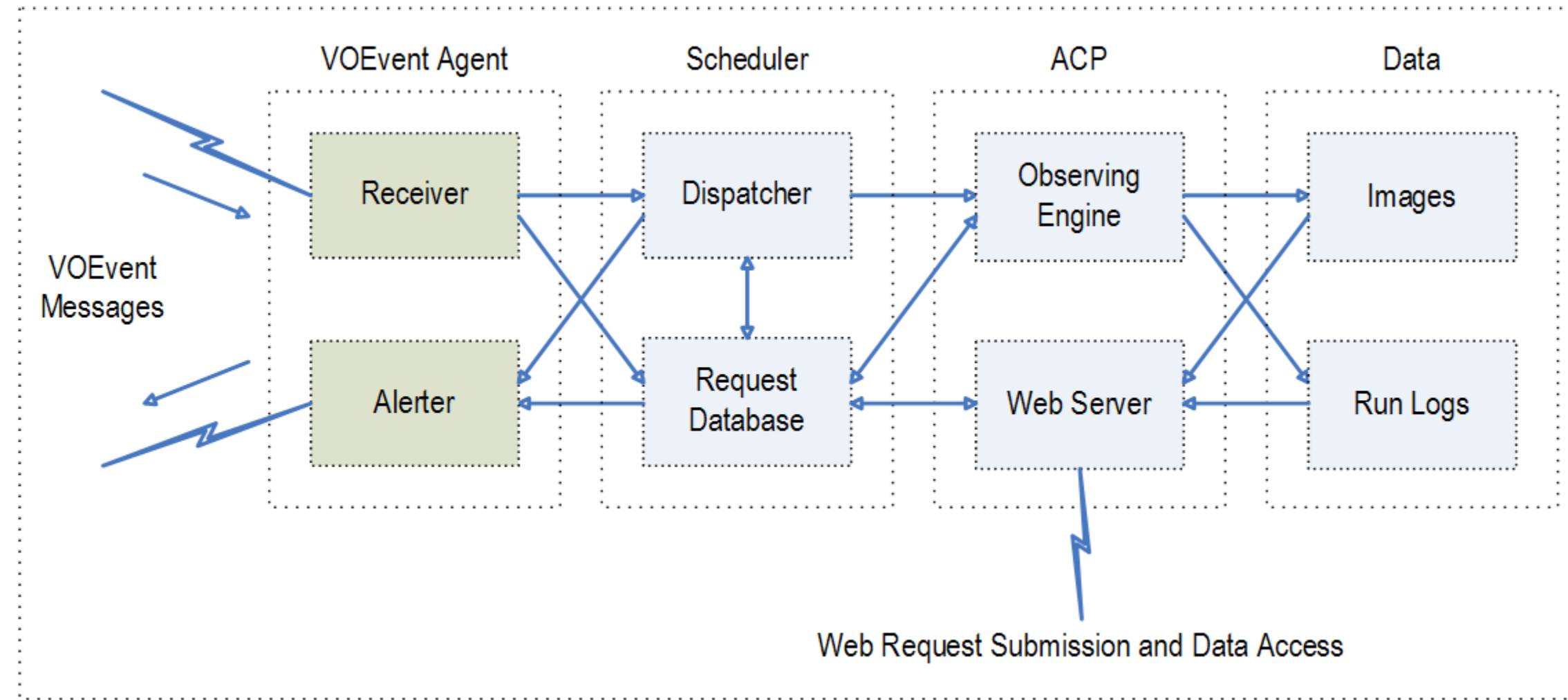
Abstract

DC-3 Dreams, SP has developed an automated VOEvent follow-up station utilizing off-the shelf hardware and packaged/supported software. The system is capable of interfacing with a wide variety of telescope mounts, imagers, filter selectors, focusers, and dome systems. It provides routine observing via a dispatch scheduler which makes “just in time” decisions as to which observing request to execute throughout the night.

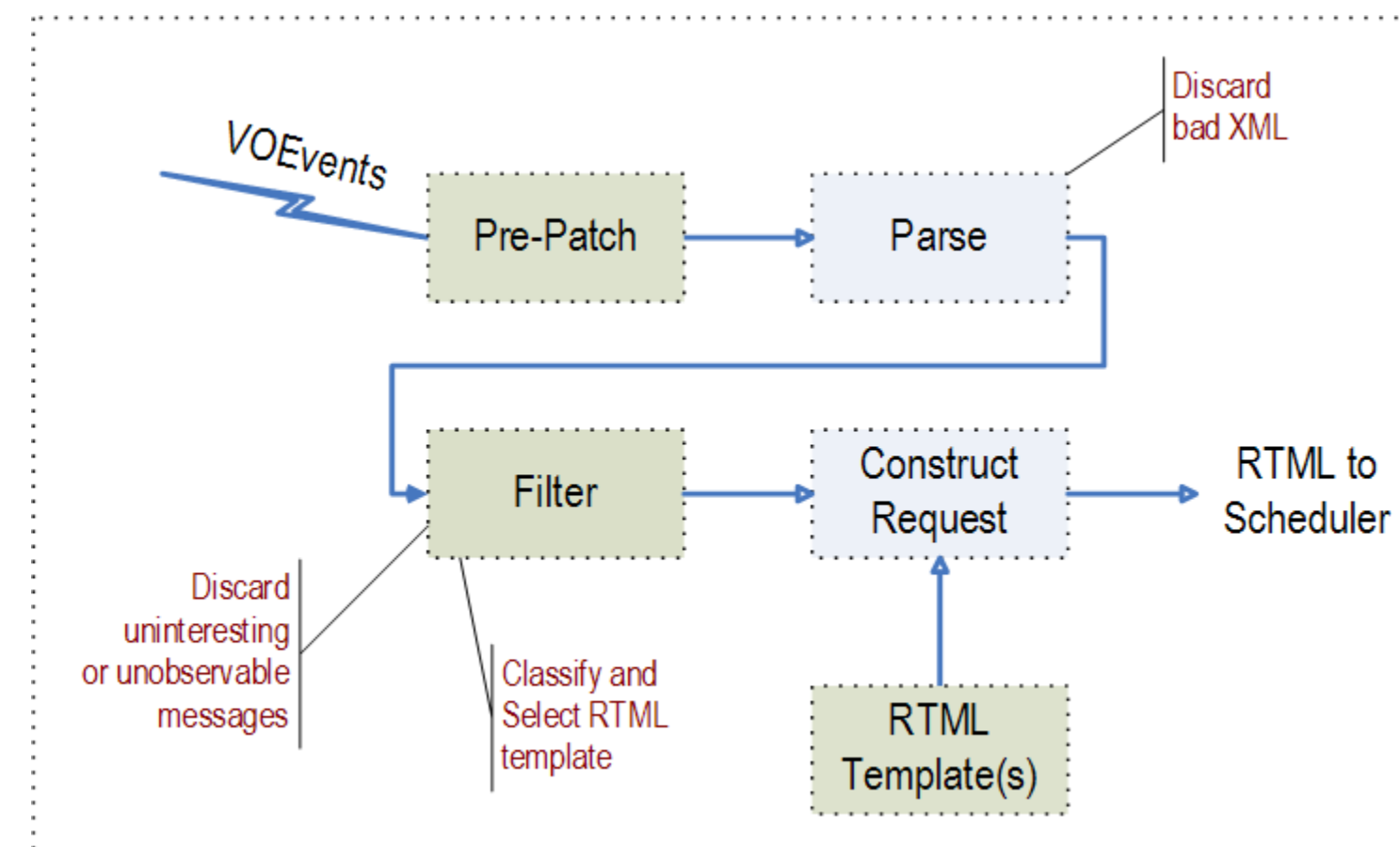
If a VOEvent trigger is received, normal observing can be interrupted in favor of a pre-programmed observing plan for the received trigger. Once the follow-up data for the trigger has been acquired, the scheduler resumes acquisition of scheduled request data according to its “next best” strategy. It is also possible to have the receiver simply queue the follow-up request (with elevated priority, for example) as part of the scheduler’s normal workload without interrupting the current request. Requests are constructed from user-specified templates for each type of VOEvent trigger.

Components

- **VOEvent Receiver/Requestor**
 - ✓ Receives VOEvent messages, filtering to identify those of interest
 - ✓ Creates request for the type of VOEvent
 - ✓ Submits request to Dispatch Scheduler and may interrupt observing
- **Dispatch Scheduler**
 - ✓ Handles observing strategy during the night, including VOEvent triggers
- **Observing Engine**
 - ✓ Executes observing requests from the Dispatch Scheduler
 - ✓ Controls all observatory components to achieve the request
- **VOEvent Alserter**
 - ✓ Sends VOEvent message advising of availability of follow-up data
 - ✓ Creates web pages containing data description and download links
- **Web Server**
 - ✓ Provides access to follow-up data (FITS images)
 - ✓ Provides detailed descriptive information on data without downloading



Overall Block Diagram of VOEvent Follow-Up Station



Incoming VOEvent Processing Flow

TITLE: GCN CIRCULAR
 NUMBER: 9211
 SUBJECT: GRB090418A Rc-band photometry
 DATE: 09/04/23 19:03:44 GMT
 FROM: Arne A. Henden at AAVSO <arne@aaavso.org>

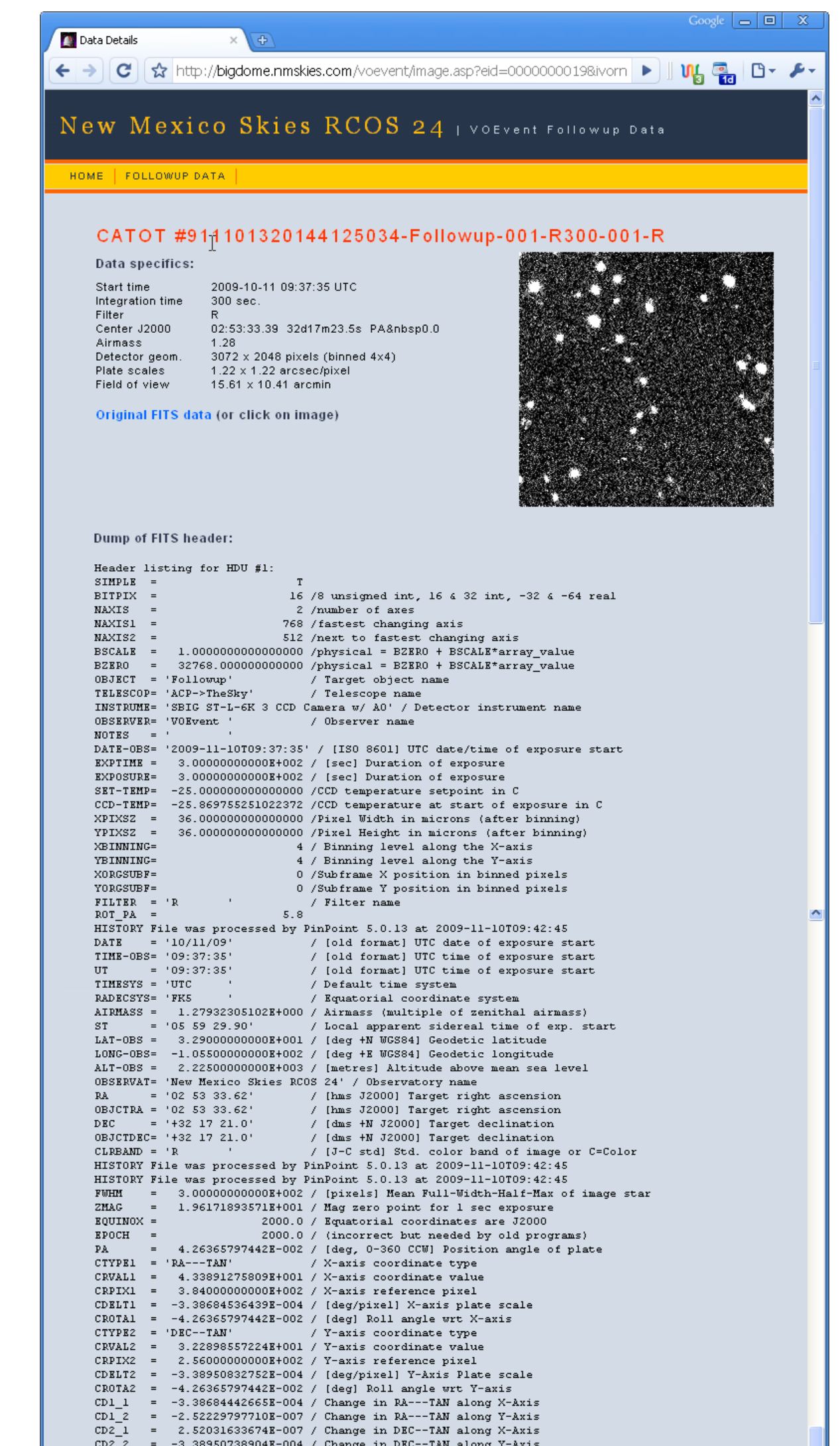
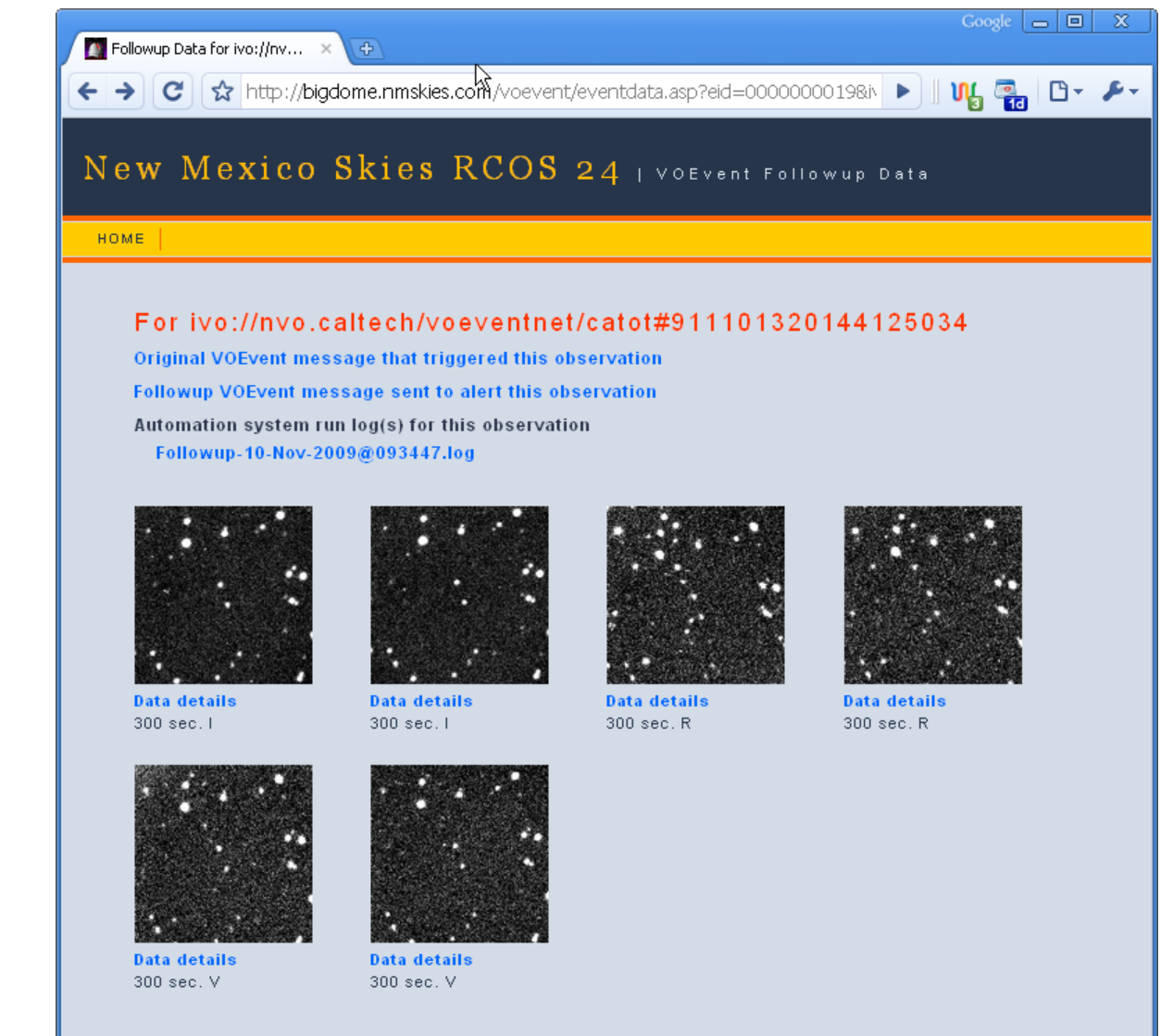
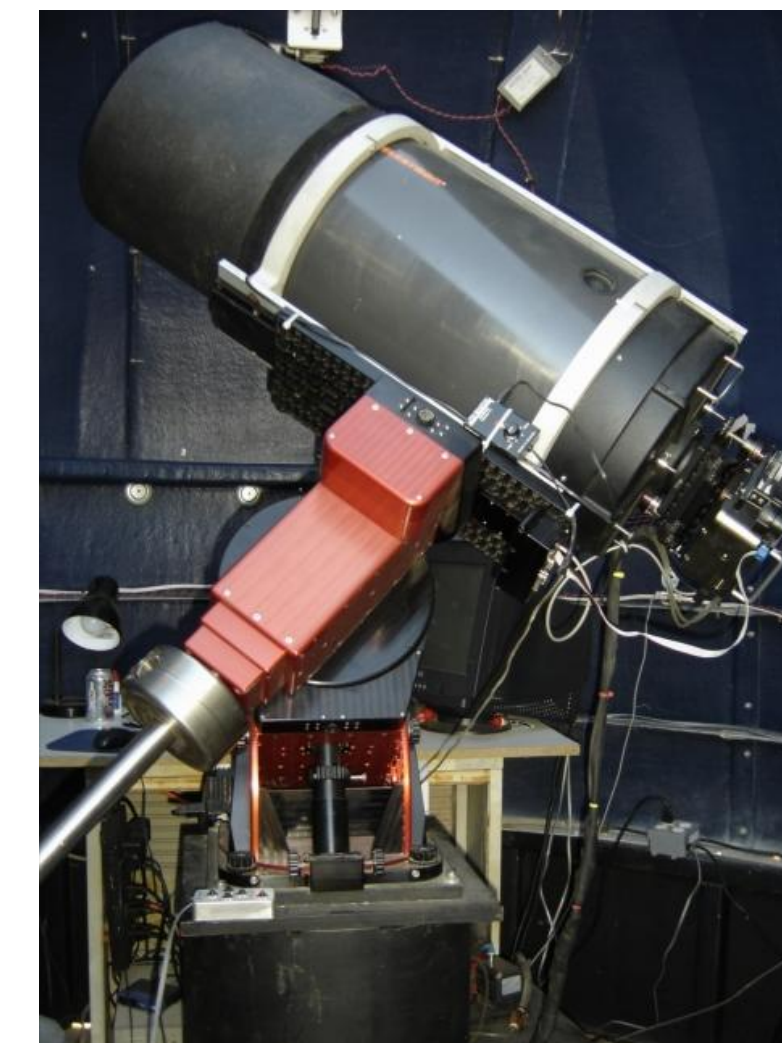
A. Henden (AAVSO), J. Gross (SRO), B. Denny (DC-3), D. Terrell (SwRI), and W. Cooney (SRO) report:

We obtained photometry of the GRB090418A afterglow reported by Chornock et al. (GCN 9148) using the Sonoita Research Observatory (SRO) 35cm telescope in southern Arizona, utilizing an automatic VOEvent trigger. The Rc-band exposures began about 2 minutes after the burst and continued until twilight. Ten 60-second and five 180-second exposures were acquired.

TITLE: GCN CIRCULAR
 NUMBER: 9220
 SUBJECT: GRB 090423 limiting magnitudes
 DATE: 09/04/24 11:36:15 GMT
 FROM: Arne A. Henden at AAVSO <arne@aaavso.org>

A. Henden (AAVSO), J. Gross (SRO), B. Denny (DC-3), D. Terrell (SwRI), and W. Cooney (SRO) report:

We observed the field of GRB 090423 (Krimm et al., GCN 9198) using the 35cm telescope of Sonoita Research Observatory with an automated VOEvent trigger for an R-band image sequence. Images began at 07:58:09UT on 2009-04-23, 170 seconds after the burst, under good conditions.



Project Status and Support

This system has been available as off-the-shelf software since late 2008. The observing engine was first released in 2001, and the dispatch scheduler was released in 2005. All software components are fully supported by the author with new releases several times per year.