SatNOGS:
Satellite Networked Open Ground Station

Daniel J. White*, SatNOGS Developers†
*Valparaiso University, dan.white@valpo.edu
†Libre Space Foundation, info@satnogs.org

Abstract—**TODO**

I. INTRODUCTION

The SatNOGS [1] project seeks to build a full stack of open technologies for satellite ground stations.

Figure Fig. 1 shows that the number of CubeSat-class satellite launches has increased nearly exponentially since the first in 2000. Previously the domain of University projects, the last 3 years have seen a huge increase in non-government or university launches. These civilian satellites include commercial, like Planet Labs' Flock-1 satellites [2], non-profit, like The Planetary Society's recent LightSail [3], and amateur, like AMSAT-NA's upcoming Fox-1 series [4].
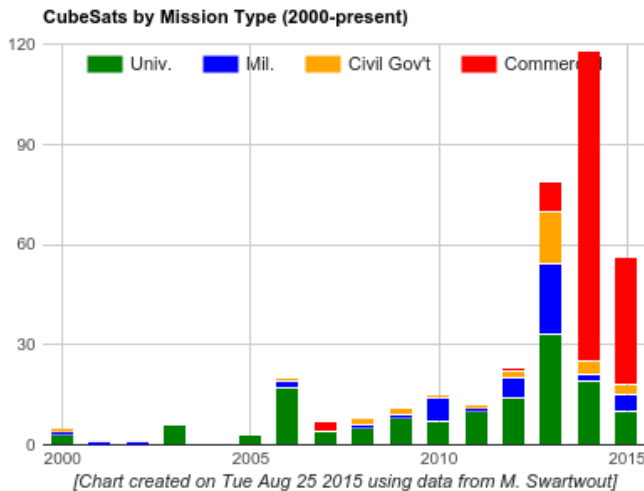

Fig. 1. CubeSat launches per year through 2015-07-17, from [5]. The "Commercial" category includes non-profit and amateur satellites.

Each satellite owner typically operates their own ground station for command and control. The low-earth orbits (LEO) of these spacecraft result in short time windows when the spacecraft is above the local horizon for communication. As a result, owners seek to enlist the help of other suitably equipped stations for collection of data. The FUNcube project is a prime example of a well-organized effort to receive data from a satellite [6].

Recent advances in low-cost, software-defined radio (SDR) technology and 3D printing have put ground station ownership within the reach of individuals. Largely composed of Amateur Radio operators, these people receive telemetry and data from many satellites and provide the information to the owners and the general public.

TODO: ground stations are not locally-used much, 1% duty cycle. SatNOGS can use the otherwise idle time. Like SETI@home, folding@home, etc.

What is missing is a civilian infrastructure to connect these many owners and ground station operators in a way that is flexible and open. The ESA's Global Educational Network for Satellite Operations (GENSO) [7] was a notable attempt at such a network aimed at University-class projects and stations. The fact that the `genso.org` domain name does not resolve to a live server is an indication of the practical current state of the project.

First proposed as a part of the 2014 International Space Apps Challenge's "Virtual Ground Station App – Global Crowdsourcing of CubeSats" Challenge [8], the SatNOGS project seeks to build such a network and community. It was later submitted as an entry to the 2014 Hackaday Prize [9], ultimately winning the grand prize of almost $200,000.

This paper seeks to give an overview of the SatNOGS project major components. Figure Fig. 2 gives a high-level overview of the relationship between users and ground stations. Figure Fig. 3 and the following sections describe the four major sub-projects: Network, DB, Client, and Ground Station. The modular approach maximizes use of already available components at a ground station.

Note, screen captures of the various software components and demonstrations are included in the more appropriate format of slides in the presentation instead of in the paper.

II. NETWORK

The SatNOGS Network is accessed by users via a web interface. The user provides details about observation that they would like to schedule (which satellite, which band, time-frame, signal encoding, etc.). From this information, the system calculates the possible observation windows from the currently available Ground Stations (GS) connected to the Network with the necessary capabilities. Once the observer confirms the proposed "observation job" then it is sent as a job to each Ground Stations' job queue to be executed. Figure Fig. 4 shows this process as a diagram.

Ground Stations then collect and send observation data back to the Network. Uploaded data is then made available to the
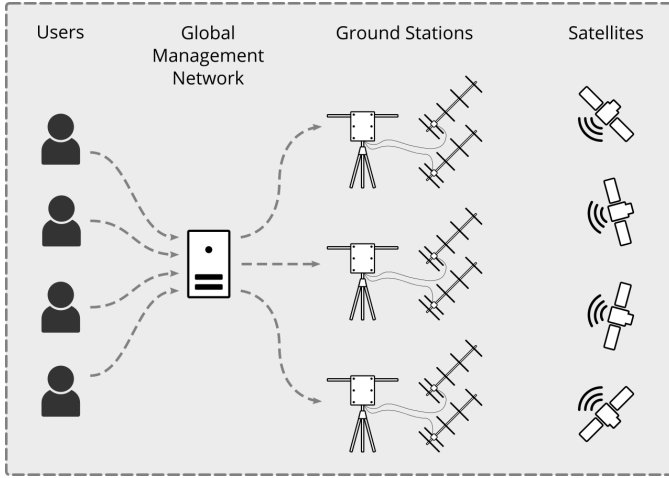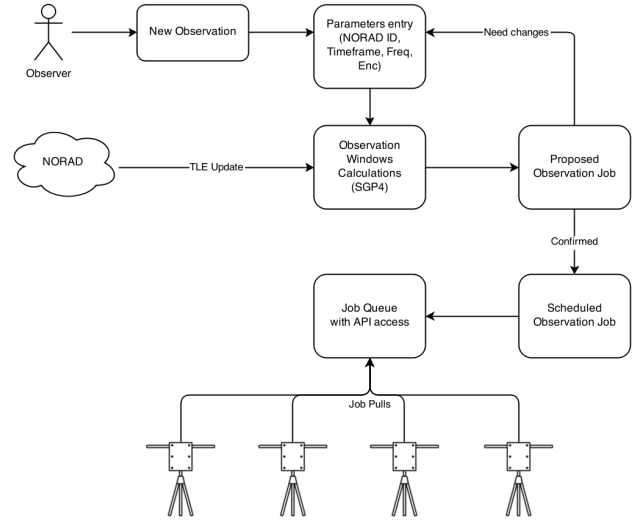
Fig. 2. Overall view of the SatNOGS concept.

Fig. 3. The four sub-projects are designed with a modular approach with well-defined interfaces, allowing the ability to relatively easily interface with existing ground stations.

initiating user and any other third party via the observation's ID. Modern web technologies are used on the Network website to provide timeline and recording visualization and playback directly in the browser. Download links are also provided for further offline analysis. No special software or installation is necessary for users to interact with the Network. SatNOGS Network provides an API to allow other applications and services to query information and, in the future, automatically schedule observations.

Calculation of candidate times when the target satellite is visible from an active ground station is performed with the assistance of the PyEphem library [10]. The library accepts orbital elements for the satellite, Ground Station locations, and the desired time frame. With higher densities of Ground Stations which can see multiple satellites, this scheduling can be optimized for many factors.

Because all code and documentation of all parts of the

Fig. 4. Diagram of a user scheduling an observation on the network.

project are free and publicly available, anyone is able to contribute to these improvements. Indeed, this open collaboration is one of the SatNOGS project's founding principles.

III. DATABASE

The centralization provided by the Network component requires a centralized source of satellite information such as frequencies and transmission modes. SatNOGS DB was created to address the fact that there was no known public source for this information. In the same spirit of the rest of the project, the database is open access and not specific only to the SatNOGS project.

Specifically, DB is a crowd-sourced suggestions app for transponder data. Satellites are identified by their NORAD (now USSPACECOM) space object catalog number and their common name. Each object has an associated set of transponder records which indicate frequencies and modulation formats. SatNOGS Network pulls this information when calculating possible observations.

Updates are accepted from the public and from other sources of satellite information with open APIs. As with the Network, user interaction is purely web-based and requires no additional software beyond a capable browser.

IV. CLIENT

SatNOGS Client consists of software running a computer which controls the ground station hardware. Figure Fig. 5 diagrams the internal (modular!) components of the client's interaction with the Network.

The poller regularly checks the Network API for observation jobs scheduled for the local Ground Station. Information contained in each job includes satellite orbital elements, receiver tuning and demodulation parameters, and timing. The PyEphem [10] library is used to calculate the necessary antenna pointing and doppler shift information for the duration
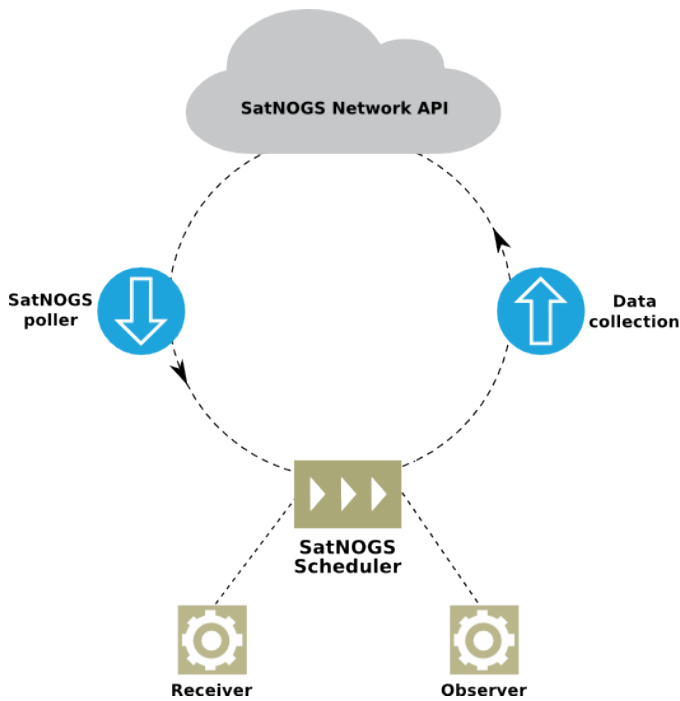
Fig. 5. Client software components and interactions.

of the requested pass. These specifics are then queued with the Scheduler to execute at the appropriate time.

V. Ground Station

VI. Conclusion

References

[1] SatNOGS — Satellite Networked Open Ground Station. http://satnogs.org

[2] Planet Labs. https://www.planet.com/

[3] The Planetary Society. http://planetary.org/

[4] AMSAT North America — The Radio Amateur Satellite Corporation. http://www.amsat.org/

[5] M. Swartwout. CubeSat Database. https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database

[6] The FUNcube Project. http://funcube.org.uk/

[7] European Space Agency. Global Educational Network for Satellite Operations. http://www.esa.int/Education/Global_Educational_Network_for_Satellite_Operations

[8] https://2014.spaceappschallenge.org/project/satnogs/

[9] Hackaday. SatNOGS wins the 2014 Hackaday Prize. http://hackaday.com/2014/11/13/satnogs-wins-the-2014-hackaday-prize/

[10] PyEphem. http://rhodesmill.org/pyephem/