Open Source CubeSat Workshop 2020

Report of Contributions

https://events.libre.space/e/4

Contribution ID: **2** Type: **talk**

r2cloud - how to build distributed ground station network alone

Saturday, 12 December 2020 14:35 (20 minutes)

In this talk I would like to give an overview of how I built r2cloud and r2server - distributed ground station network. I will cover the key design and organizational decisions that worked and didn't.

The key topics are:

- Why open source and what it would mean from organizational point of view.
- Key design and technologies that help running r2cloud alone.
- Why it is important to define what project is and what project is NOT?

Primary author: Mr RODIONOV, Andrey

Session Classification: Room #2

Contribution ID: **3** Type: **lightning talk**

SDNx DebriSat: Detecting Milli Space Debris In the Low Earth Orbit

Sunday, 13 December 2020 16:30 (4 minutes)

About half a century's space programs have littered the Earth's orbit with more than 22,000 man-made objects of a size larger than a softball and hundreds of thousands of tiny fragments. This increasing number of space unused debris are causing serious threats to the present and future outer space activities, in area of both government (military, intelligence, and civil) and commercial.

Experts presume that the current population growth trend will be predominantly influenced by catastrophic collisions that are likely to occur every five to nine years.

To curtail the expanse of man-made debris scattered from its space launches and inactive satellites, our group of Indian Students are developing 'SDNx DebriSat' – to launch a constellation of NanoSatellites for as an early warning arrangement to avoid threats and other hazards to Indian satellites.

SDNx DebriSat will track 1,000,000 debris of size less than 1 cm is our targeted fragments with a sensor range of 600kilometres.

Primary author: YADAV, Govind (SDNx India)

Session Classification: Room #1

Contribution ID: **4** Type: **tutorial**

Restful Data Storage and Visualization

Sunday, 13 December 2020 16:45 (45 minutes)

For operation of a satellite mission (and other projects) it is a very common need to store and display large amounts of data. We can group such data in two categories: planning data and timeseries data. Planning data typically consists of records that contain time ranges and ancillary information (such as for booking of groundstation contact times), where as timeseries data is composed of time-stamped datasets (such as from onboard measurements).

There are certainly many different software applications (open source and proprietary) available to tackle those issues. What we have created however, is an ecosystem that is composed of microservices with REST APIs, exclusively developed in Python3 and flexible to build various kinds of system applications.

In brief, we have developed a timeseries datastore and a documents store that provide REST API interfaces to store, manipulate, and query large amounts of data. A neat Python rest interface provides a pythonic interface to those services. Finally, a data visualization library that uses Vega-Lite inspired syntax and supports interactive plotting makes it convenient to plot those data.

In this tutorial I will show you how to use those Python modules for storing your large amounts of data and to display it.

Primary author: Mr SCHOLZ, Artur (LibreCube Initiative)

Session Classification: Room #2

Contribution ID: **5** Type: **lightning talk**

Power Conditioning and Distribution Unit for 1U CubeSat

Saturday, 12 December 2020 16:10 (4 minutes)

The Power Conditioning and Distribution Unit (PCDU) as an essential part of a Electrical Power System (EPS) of each satellite. It's purpose is to regulate the power coming from solar panels and to manage batteries in order to supply constant power as needed to the satellite electronics.

We have embarked in developing an open source PCDU to be used in 1U (single unit) CubeSats (or larger), that is simple, robust, and reliable.

In brief, the requirements placed on the PCDU are:
- regulate input power from solar panels (up to six connections)
- interface to a Li-Ion battery module, that may uses two Li-Ion cells in parallel to increase capacity and for redundancy.
- integrate system power switch to power off all outputs (e.g. using Kill Switches)
- provide stable 5 Volt output with at least 5 Watt
- provide switchable 5 Volt outputs
- integrate a telemetry and telecommand interface (TMTC IF) to control and monitor the PCDU
- have this TMTC redundant, and switchable via pulse on a select line
- have all modules and units of the PCDU operate in redundancy, such that there is no single point of failure

As one can immediately see, only 5 Volt is provided as power output. This is done purposely, as 3.3V or 2.5V (also used often, such as for on board computer), can be easily drawn from this 5V line via simple low dropout regulators, at those systems that need it. This way, it simplifies the PCDU a lot, and focus can be put on making this 5V line as robust as possible.

In this presentation we present the current status of the developments and show how you can get involved.

Primary author: Mr SCHOLZ, Artur (LibreCube Initiative)

Session Classification: Room #1

Contribution ID: **6** Type: **tutorial**

Demo on writing/loading/installing/executing your code on OPS-SAT

Sunday, 13 December 2020 16:45 (1h 15m)

OPS-SAT is a mission that runs a system on module with ARM-9 processors and an FPGA (Cyclone 5) coding environment. The ARM has a Linux Ångström distribution with Python3, Java, C++, and C available. On top of that ESA is providing a framework that experimenters can use to deploy on-board experiments in a App-like way. All this is available for experimenters to use for free until Nov 2021 and it is flying now. The question is how does it work practically? In this workshop the OPS-SAT flight control team will walk you through some user stories pretending to be experimenters from the idea stage to actually deploying the code on the flatsat and executing it. This will be done live.

Primary authors: EVANS, David (ESA); Mr MLADENOV, Tom (ESA); MARSZK, Dominik (European Space Operations Centre); Mr LABRÈCHE, Georges (European Space Agency); SHIRADHONKAR, Vasundhara; GAGLIARDINI, Lorenzo

Session Classification: Room #1

Contribution ID: **7** Type: **lightning talk**

Code in Space

Saturday, 12 December 2020 16:22 (4 minutes)

Code in Space is an international opportunity where students from around the world can submit proposals to upload their code to a nanosatellite orbiting the Earth. The code will be executed by the satellite's onboard computer and will be tested under real space environment conditions. The code executions test results will be transmitted back to Earth via a satellite ground station.
GOALS:
-To increase awareness about current opportunities and challenges in the satellite industry
-To encourage solutions to current challenges faced by the satellite industry
-To encourage and empower students to design and develop new concepts that could contribute to the advancement of satellite technology
-To increase awareness about new space (space 2.0) revolution and its potential impact on humanity
TOPICS:
-Satellite Communications
-Satellite Operating System (Flight Software)
-Ground Stations & Ground Operations
-Scientific Investigations (based on onboard sensors readings such as temperature, magnetic field, acceleration, and light)
-Games and Apps. using Satellite Signals or Data

Primary author: ALFEELI, Bassam (Orbital Space)

Session Classification: Room #1

Contribution ID: **8** Type: **lightning talk**

Portable QO-100 Software Defined Ground Station

Sunday, 13 December 2020 16:14 (4 minutes)

I would like to talk about my ongoing project "Portable QO-100 Software Defined Ground Station" which is being documented in "https://github.com/barisdinc/QO100-Partable"
The main idea is :
- As cheap as possible
- As small as possible
- As open as possible
- PORTABLE

Based on Arduino Nano uSDX as SDR TRX as IF transceiver at 50MHz, it has dual receivers to have lock to beacons capability for frequency stability. Transverter and power amplifier stages are all based on open hardware designs.

The aim of the project is;
- to design a "GEO satellite base station" in a collabrative environment
- learn SDR TRX basics and maths behind it
- a portable station that can be used in EMERGENCY conditions

During the talk and/or presentation I would like to talk about the basic design concepts and design decisions of the system. Also I would like to invite everybody to contribute to this project with their ideas and workforce.

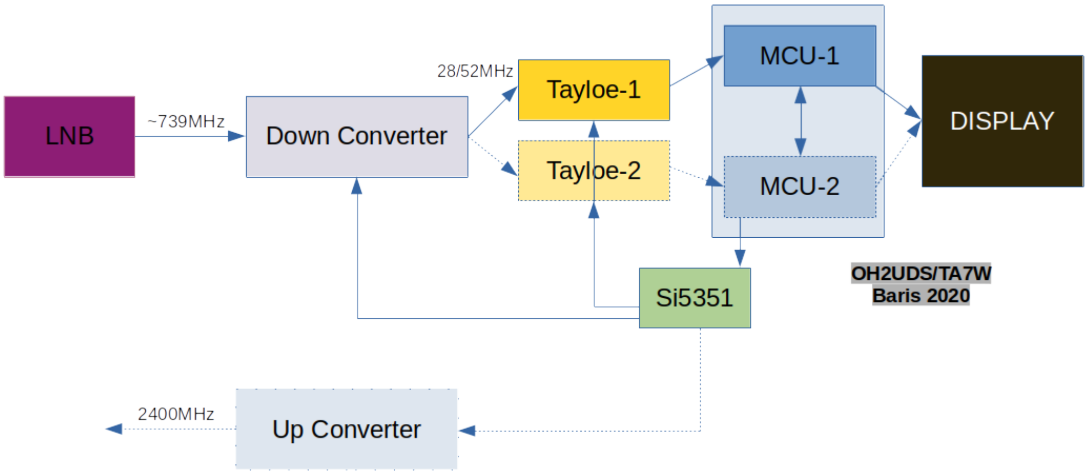

Figure 1: enter image description here

Primary author: DINC, Baris

Session Classification: Room #1

Contribution ID: **9** Type: **talk**

Orbit.M - An Open-Source Software for LEO Orbit Maintenance in Python

Sunday, 13 December 2020 14:50 (20 minutes)

Indico rendering error

```
Could not include image: [404] Error fetching image
```

What is Orbit.M?

Orbit maintenance analysis tends to be rather rigorous with commercial software like STK, or tedious if you have to develop your own simulator in code. Orbit.M offers to the OSCW audience an open-source, graphical, easy-to-use, and free orbit maintenance simulator and propulsion sizing tool, meant for circular orbits, for anyone and anywhere.

It is a software with a GUI where you can fill in your orbital parameters, satellite masses, satellite areas, drag parameters, and the software will compute your desired Delta-V necessary for the mission, sizing it against your choices of thrusters (based on the Isp).

How do I use Orbit.M specifically?

First, check that you have Python 3 installed, and Python standard libraries: **TKinter, NumPy, Matplotlib**, and among others (**os, datetime, comtypes etc**). If you do not have Python, I recommend using the Anaconda installer, and running it on Spyder 4 simply because that's what I'm using right now. At the time of the latest commit and push, NumPy version is 1.18, and Matplotlib is 3.1.3. Older versions should work fine as ORBIT.M doesn't use any of the newer features.

The software can be started by running the Python file **orbitm.py** in the main directory (which is equivalent to the directory you see on the master branch on ORBITM's github page). You should see a GUI that looks like the one below, pop up:

Indico rendering error

```
Could not include image: [404] Error fetching image
```

You can fill in your spacecraft and orbit parameters through the GUI.

Next, before you run ORBITM, you should open up the "thruster_shortlist.txt" file on the main ORBITM directory, and fill in any thrusters you wish to size your missions against. Some thrusters I had previously been aware of are written inside this shortlist as an example. The purpose of this shortlist is to graphically compare its Isp and fuel capacity to the suitability of your mission later on.

Do note that the option to use STK10 and STK11 requires a valid legal license for STK Astrogator and STK Integration.

Now, you can run ORBITM.

What can I expect as the output of Orbit.M?

Orbit.M would output three plots - one of altitude decay, one of the mean semi-major axis decay, and finally, your Delta-V sizing profile. The program will also output a time-schedule for orbit maintenance in the main directory of ORBITM, as a text file "deltaV.txt". For the circular orbit @ 500km, with a tolerance band of 5km, an example plot is shown below (blue plot being Sam's simulator, and orange plot generated using STK10):

Indico rendering error

```
Could not include image: [404] Error fetching image
```

What else is expected from Orbit.M?

The goal of sharing Orbit.M for the OSCW 2020 is to give satellite and space enthusiasts some simple tools to begin orbit maintenance calculations - especially for aspiring students who cannot afford the time-effort nor money for expensive space simulator software. The accuracies of my model are cross-checked with that of STK's as you can see in the chart above.

In the future, I hope to welcome users experienced in NASA's GMAT program to automate orbit maintenance routines uing Python as a 4th simulator option in Orbit.M. If you like open source, and you enjoy orbital mechanics, do consider adding your contributions here for other GMAT users too!

Looking forward to seeing everyone in OSCW 2020!

Primary author: LOW, Samuel

Session Classification: Room #1

Contribution ID: **10** Type: **lightning talk**

'New comer' ground station challenge : demodulate data

Saturday, 12 December 2020 16:14 (4 minutes)

The NewSpace era has seen the rise of various new actors, especially in the field of telecommunications. For example, the *SatNOGS* network (SatNOGS website) allows people from around the world to receive data from CubeSat missions with their own ground station, as an open source project.

In this context, the CCERES team (Paris Observatory) is a 'new comer': we operate a ground station that has been part of the *SatNOGS* network for more than a year. The station has successfully received and demodulated thousands of 'waterfalls' and frames from many CubeSats. However, since the recent launch of *AMICal Sat* (AMICal Sat's website), the ground station has been unable to demodulate any of his transmissions, while perfectly receiving and forwarding raw frames to the network. Today, we are looking for a solution to solve this isolated case (that might not be so singular after all …).

We will explain in this short talk what has been done to identify and understand the problem. Then, what is currently under investigation, future solutions and an overall feedback will be shared.

Primary authors: LE LEUCH, Louis (CCERES, space pole of PSL Université, hosted at Paris Observatory); Mr ZIDA, Romain (Paris Observatory Student); Mr DJIBRIL DRAMANE, Souwédou Silla (Paris Observatory Student)

Co-author: Mr SEGRET, Boris (CCERES, space pole of PSL Université, hosted at Paris Observatory)

Session Classification: Room #1

Contribution ID: **11** Type: **talk**

An Open-Source on-board computer platfom for CubeSats

Sunday, 13 December 2020 14:50 (20 minutes)

We will present an ongoing project of hardware development for CubeSat applications. Vision Space Technologies has been working on this project known as VST104 since mid 2020. Our goal was to design and open-source a small family of hardware boards in the PC104 format. The effort was aimed to onboard computers (OBCs) and their development & testing auxiliaries. At the moment, a single MCU board has been printed, soldered, and successfully validated. A FlatSat is also included for testing. Internally, this project's outcome will be a platform for testing mission control systems (MCS), and eventually, new algorithms with a future extension of an onboard field-programmable gate array (FPGA) technology. For now, we handle this project as a contribution to the LibreCube initiative and as a support of the local university club TUDSaT.

At the workshop, we would like to talk briefly about this project and to present the significant members of our new PC104 CubeSat family:

Board Sierra - Single redundant OBC board. This module is driven by the STM32L496 microcontroller and combines many exciting features. Our goal was to fulfill most of the space industry requirements while keeping the cost down and manufacturing easy. A significant number of redundant peripherals with separate isolators are available for communication with other CubeSat modules. With redundant external memories and precise clock sources, this module can handle various user-case scenarios. A robust power management circuitry, a group of temperature sensors, and other safety measures contribute to the reliability of this module. Of course, all of the components are military rated with automotive certifications.

Board Delta - Double redundant OBC board. This module implements Board Sierra in full double redundancy. The original Sierra module was extended with its own mirrored copy resulting in two independent OBCs on the same PC104 module. The potential of this board is in user scenarios where reliability is essential. In case of a system fraud, a supreme logic can simply switch between the identical OBCs sharing (almost) the same software and electrical characteristics.

Element Foxtrot - FlatSat with external power module. This auxiliary board serves as a development and show-off board. The flatsat is capable of hosting four PC104 modules with LibreCube PC104 header pinout. A breakable part serves as an external power module driven from a power adapter or USB-C (handshake is supported). Element Foxtrot is an ideal tool for testing and developing different CubeSat modules, temporary replacement for power distribution unit (PDU), and a nice way to present already developed modules.

As we mentioned, this VST104 project is still active, with a group of exciting milestones steadily approaching. Therefore it would be our pleasure to contribute to your workshop and to inform our community about it.

Primary author: GEIB, Filip (VisionSpace Technologies GmbH)

Co-authors: FEITEIRINHA, Jose (VisionSpace Technologies GmbH); ÖZLEYEN, Selman

Session Classification: Room #2

Contribution ID: **12** Type: **lightning talk**

Phoenix - An open source orbital cloud infrastructure

Sunday, 13 December 2020 16:10 (4 minutes)

Phoenix is a project launched last October as part of the Federation initiative.

The Phoenix project aims to design, build and put into orbit the first bricks of **an orbital cloud** (a shared IT infrastructure in orbit). This cloud will be based on a decentralized and modular architecture.

Any organization on a global scale could contribute to its deployment by incrementally adding **standardized and open source bricks** to the infrastructure (space or ground segment).

It aims at providing **space-to-space and space-to-ground IT container services**, which could be put to multiple uses (Image computation, trajectory computation, signal processing…).

Its main components are:

- **IT architecture**: each satellite will constitute a "node" of the infrastructure, similar to the physical components of a terrestrial IT infrastructure, mainly an information processing and/or storage node (server), and a communication node (switch).
- **Space platform**: hardware design of what is specific to a server / switch in orbit: power supply, protection against environmental hazards (temperature, radiation, debris …), attitude control, possibly propulsion. Potentially we could consider the virtualization of the on-board computer (no dedicated physical on-board computer).
- **Communication system**: intra-constellation communication and communication with the ground.
- **Ground Segment**: definition of the operating modes of our satellites,
  the link with other space actors (tracking, collision detection, various interferences…).

Currently the project is at the early design phase. We are using the model-based system engineering methodology to perform an operational analysis, and from there deep dive into sub-systems.

Our **current roadmap** includes :

- Testing the early software layer from an existing orbital platform (ISS, satellite…) in 2021
- Launching a 2U proof of concept in 2023
- Launching the first nodes of the operational constellation in 2026

And we are looking for all interested contributors to join us to make this sky-reachng project real!
No skill nor diploma required, only passion and the will to dedicate time and energy to moving the project forward.

Primary author: HARTMANN, Damien

Session Classification: Room #1

Contribution ID: **13** Type: **talk**

QUBIK Twins: The unplanned mission of an unbuilt satellite to be integrated into a deployer that does not exist

Sunday, 13 December 2020 15:30 (20 minutes)

Qubik 1 and Qubik 2, are 2 PocketQube open-source satellites, designed, developed, integrated and delivered together with a PocketQube deployer (PICOBUS) to be flown as part of to participate in the DREAM Payloads Program of the inaugural launch of Firefly Alpha rocket.

The Qubik twins are tasked to perform Launch and Early Operations Phase (LEOP) Satellite Identification and Tracking experiments. PicoBus will deploy several 6 PocketQube satellites (includingt the two Qubik satellites).

The presenter will focus on the challenges of the parallel development of the Qubik pico-satellites and the PicoBus deployer systems in a limited timetable while the Covid-19 pandemic loomed.

Primary author: Mr DARADIMOS, Ilias (Libre Space Foundation)

Session Classification: Room #2

Contribution ID: **14** Type: **talk**

Cronos sounding rocket

Saturday, 12 December 2020 14:55 (20 minutes)

Cronos is a hybrid-fueled sounding rocket engineered by the student team White Noise in close collaboration with and under the auspices of Libre Space Foundation. It is a project designed and manufactured in an open and participatory manner.
Cronos is designed to lift 4kg of deployable, payload mass, housed right below the airframe's nose cone. Its paraffin wax and nitrous oxide hybrid propulsion system is able to produce an estimated 3000N of thrust for approximately 3s. Thus, it is estimated to reach mach 0.88 and an apogee of 3km AGL. The sounding rocket is designed so it will be safely recovered after the launch. In order to achieve this when at apogee, the recovery sequence will begin with the ejection of the drogue parachute followed by the main parachute at the appropriate altitude. The two parachutes will be ejected by utilizing an in-house built CO2 system, so as to safely recover the full rocket, making it thus, a 100% reusable sounding rocket.
It's advanced materials make Cronos lightweight, with a dry weight of only 21kg and a length of 4m.
Cronos is the effort of a team of University students to manufacture a sounding rocket so as to be able to explore and identify the challenges of such an endeavor. The students have familiarized themselves with the processes and operations of following open development and open-source approaches. The primary aim of the project Cronos is to become the test platform for exploring future possibilities of building more advanced rockets, capable of reaching higher altitudes, designed and manufactured in a collaborative and participatory way.

Primary author: PATSAS, Thanos

Session Classification: Room #2

Contribution ID: **15** Type: **talk**

SatNOGS - State of the Union

Saturday, 12 December 2020 14:15 (20 minutes)

An overview of the SatNOGS project, a network of satellite ground stations around the world, optimized for modularity, built from readily available and affordable tools and resources.

Low Earth Orbit (LEO) satellite launches rate increases with the participation of old and new entities. In this growing environment SatNOGS provides a scalable and modular solution to track, receive telemetry, monitor and command & control satellites.

SatNOGS global community, dedicated to its free and open source values, develops hardware ground station designs (antennas, rotators, electronics), software for SDR-based communications, satellite scheduling and mission monitoring platforms.

SatNOGS continuously develops and improves its infrastructure to allow observers use this networked ground segment and remotely operate SatNOGS ground stations around the world. It provides also an easy way to store, access and view increasingly received satellites data, by supporting VHF and UHF bands and by moving forward with higher bands like L, S and X.

This is a proposal for a "state of the union" talk about SatNOGS focusing on what has happened since previous OSCW, growth, development and trajectory for features and expansion.

Primary author: DAMKALIS, Alfredos Panagiotis (Libre Space Foundation)

Session Classification: Room #1

Contribution ID: **16** Type: **talk**

DOCKS – An open-source software suite for space mission profiles

Saturday, 12 December 2020 14:35 (20 minutes)

The growing pace of the NewSpace has resulted in a significant rise in space missions, and consequently, a need for mission profile tools. Though many tools already exist in the market, either they are not capable of doing the complete mission profile or have high costs, which make them unsuitable for many. To overcome this gap, an open-source software suite targeting scientific nanosatellite mission profiles called DOCKS is developed at CCERES, the space pole of PSL Université, hosted at Paris Observatory. An additional advantage of DOCKS is that it is compatible with VTS, the French space agency (CNES) software to visualize and animate satellites in a 3d simulation.

DOCKS will be composed of seven thoroughly validated Python modules as plug-ins, which can also be used as standalone. Until now, the deep space trajectory Propagator, Energy Power Simulator, and Intervisibility modules have been developed. These are regularly updated on public GitLab with new features and improvements based on user feedback. Currently, DOCKS is being restructured to make it faster by adopting methods like parallelization or remote service from our servers. The latest developments in DOCKS with our future perspective will be presented in the talk. Further, CCERES also provides Model-Based System Engineering (MBSE) support to various CubeSat projects for the early phases of mission profile modeling, so the use of DOCKS in this context will also be demonstrated.

Primary authors: Ms JAIN, Rashika (CCERES, space pole of PSL Université, hosted at Paris Observatory); Mr DELRIEU, Thibault (CCERES, space pole of PSL Université, hosted at Paris Observatory); Mr SEGRET, Boris (CCERES, space pole of PSL Université, hosted at Paris Observatory)

Session Classification: Room #1

Contribution ID: **17** Type: **talk**

Nanospace, an open source tool to help concurrent engineering teaming in cubesat preliminary design

Sunday, 13 December 2020 14:30 (20 minutes)

Designing a cubesat requires close interrelation between different fields, with respectively strong level of expertise, all the more so as development progresses.

Nanospace is a tool designed to ease interaction between expert tools and a graph database. It is not a heavy client application, but a web oriented one. The front-end, Nanospace-UI, an Angular GUI, is accessible through a browser. Data management is handled by a graph Neo4j database deployed on the back-end server side. This database is able to centralize all the project preliminary design required data, taking into account concurrent access of all experts.

Key contribution of Nanospace to the community of Concurrent Design Engineering tool is a way to ease inter-connection of experts (and their tools) around project data. In a pragmatic way and from each subsystem engineer point of view, it allows to link in a simple and automated way their own expert tool to a common, unified and updated database at the system engineering level. Let say that if mission analysis changes orbit altitude, effects should be propagated to all subsystems, should be taken into account for the mission requirement, and should lead to iterating and converging over new sets of output data. Three way of connecting the expert tools to Nanospace database are provided:

- Manual interaction with NSS Web User Interface - the "shared spreadsheets style";
- Embedded code to ease interface - an Angular component and a python API are provided to ease code integration;
- Full compliance with the Nanospace REST API - allowing you to develop your own independent service application directly interacting with the database.

Nanospace is the backbone of a larger project, NSS (Nanostar Software Suite). NSS aims at helping to get a strong data consistency between expert software during a mission analysis preliminary design. Goal is to provide a set of open-source tool that can be used for designing each subsystem of a cubesat, interconnected through a centralized database. That should let the choice for the user, depending on its skills and tools, the best way to fit his project needed.

Nanospace is currently under development.Nanospace is Open source (AGPLv3 Licence), a prototype is available 1. A test server is also deployed (2). Part of this work is supported by SUDOE Nanostar project 3.
1 https://gitlab.isae-supaero.fr/nanostar/nanospace
2 https://dcas-nanostar/
3 http://nanostarproject.eu/

Primary authors: GATEAU, Thibault (ISAE-SUAPERO); Mr SENANEUCH, Lucien (ISAE-SUPAERO)

Session Classification: Room #2

Contribution ID: **18** Type: **talk**

OLLO - A 3U Open Access astronomical telescope mission

Saturday, 12 December 2020 15:35 (20 minutes)

Open Little Luminary Observatory - OLLO is a small orbital telescope system concept aimed to provide an open access platform for astronomical observations. OLLO's mission is to create an open, interactive tool allowing everyone to access and request images of astronomical objects, such as planets, nebulae or star clusters. This concept aims to bring astronomy and science closer to the public. Our main objectives are to:

1. Complement existing astronomy infrastructure with a universally accessible observation platform for educational, scientific and amateur purposes.
2. Design a 1.5U optical imaging telescope to be integrated into a 3U CubeSat platform.

To achieve these goals we will connect the CubeSat observation platform with a web service through the available ground segment at the University of Vigo. This will allow users to access previously collected data or to request observations of specific bodies, which the OLLO orbital platform shall carry out when the target object is in sight of the payload. After the observation, the resulting data is downloaded to the ground segment and is made accessible to the end users.

The motivation for our second mission objective is derived from most CubeSat-based astronomical telescope missions using 6U platforms. Using past missions data, especially JPL's ASTERIA mission, a feasible target resolution of 15 arcseconds per pixel can be achieved with the corresponding pointing accuracy and pointing stability requirements, which should allow for wide field of view images of deep-sky objects, like M31-Andromeda, the M81 and M82 Galaxies, or M42-Orion Nebula.

The OLLO mission was selected to participate in the Concurrent Engineering Workshop organized by the ESA Academy. Several members from UVigo SpaceLab will attend the workshop when it takes place in 2021 at their installations of ESEC-Galaxia in Belgium to further develop the concept.

Primary authors: Mr DRAGOS DARAU, Vlad (University of Vigo); Mr CAMANZO MARIÑO, Alejandro (UVigo SpaceLab); Mr VALE XOUBANOVA, Manuel (UVigo SpaceLab); Mr CHEDAS CASTRO, Pablo (UVigo SpaceLab); Mr DIZ FOLGAR, Manuel (UVigo SpaceLab); Mr CARRO PARAFITA, Pedro Manuel (UVigo SpaceLab); Mr SALGUEIRO VIDUEIRA, Isaac (UVigo SpaceLab); Mr CAMBÓN PERISCAL, Ana (UVigo SpaceLab); Mr CASANOVA ÁLVAREZ, Marco (UVigo SpaceLab); Mr NOVO REGO, David (UVigo SpaceLab); Dr NAVARRO MEDINA, Fermín (University of Vigo); Dr AGUADO AGELET, Fernando (University of Vigo)

Session Classification: Room #2

Contribution ID: **19** Type: **talk**

Charting the development of the Open Source Space Ecosystem

Saturday, 12 December 2020 15:15 (20 minutes)

Having witnessed the impact that an Open Source approach has made in many segments of the software industry, numerous teams around the world are embracing the Open Source philosophy to implement systems throughout all segments of the space industry and adjacent sectors.
Open Source Space systems have the potential to change paradigms about how space missions should be implemented; to transform the future of the space industry; to truly democratise space by reducing barriers to access, and to open up the benefits of "Future Space" space to society.
From components and subsystems through to operations; from small teams working on small high altitude missions through to global collectives with a vision of interplanetary settlement; from training and education platforms through to commercial systems; delivering applications as diverse as biological experiments through to space situational awareness; there are a multitude of teams working to create open systems and applications that all can contribute to and benefit from.
This presentation will provide a consolidation of the outputs of the workshop session we held at OSCW'19 and seeks the contributions of the OSCW'20 participants to evolve and generate an up-to-date snapshot of the Open Source Space capability landscape.

Primary author: BERNIE, Anita (KISPE Space Systems Limited)

Co-author: Dr PAFFETT, John (KISPE Space Systems Limited)

Session Classification: Room #2

Contribution ID: **20** Type: **talk**

Progress and Developments in Open Source Electric Propulsion at Applied Ion Systems

Saturday, 12 December 2020 15:15 (20 minutes)

The state-of-the-art in the field of open source space technologies has continued to progress rapidly in recent years. While significant advances have been made, there are several key areas where accessibility remains a challenge. One of these critical areas requiring further focus on in the open space hardware area is that of propulsion, which still remains prohibitively expensive for most nanosat and picosat teams, and with often high complexity and incomplete design details in academic literature, further increases the barrier of entry for development. Electric propulsion technologies in particular offer unique advantages specifically for Cubesat and smaller class satellites for a wide range of applications, from attitude control, to debris avoidance, station keeping, and other orbital maneuvers.

Since the development of the first fully integrated open source micro-PPT at Applied Ion Systems (AIS) last year, the AIS-gPPT3-1C, currently aboard the AMSAT-Spain GENESIS N and L PocketQubes, several new prototype systems have been tested through the AIS effort. These systems include new variations on the gPPT3 thruster design, as well a new PPT design, an ionic liquid ion source (ILIS) electrospray thruster, and further research into additional higher-power conventional technologies such as RF gridded ion thrusters, RF plasma thrusters, and micro-resistojets. AIS is also currently engaged in R&D of new highly experimental propulsion systems and alternative fuels for plasma and ion thrusters, covering a range of satellites from PocketQubes to large Cubesats and beyond, in power classes ranging from a few Watts to hundreds of Watts and up. The goal of these developments are to provide the community with a way to access these exciting technologies, allowing for them to further expand mission capabilities, while radically reducing the cost of development and systems as a whole.

For prior developed systems, current designs in testing, and new future designs in the works, AIS has also adopted the new CERN Open Hardware License V2 for releasing design files to support its mission for open source electric propulsion.

Primary author: BRETTI, Michael (Applied Ion Systems)

Session Classification: Room #1

Contribution ID: **21** Type: **lightning talk**

The OpenSource in Space Podcast

Sunday, 13 December 2020 16:18 (4 minutes)

Open-source projects are great but discovering new projects is not that easy and keeping up-to-date with existing projects on a deeper level takes time and often a founded understanding of the project is necessary since these teams lag dedicated PR departments.�But what if a central hub would exist where the makers themselves could explain their ideas and thoughts and - even more convenient - if I could get this information when my brain runs in idle anyway: during my regular workouts? While I cook or clean the dishes? While I commute to work?

I want to launch an interview podcast to make open-source space projects more visible for the community and to provide a platform where the projects can talk about big updates for the community in an easily reachable way.

In my presentation I want to talk about the following topics:
- the general idea of the podcast
- the benefits why anyone should participate
- the general structure of each episode
- the general plan for the future

Current Website of the project:
https://jan-peter.space/os2

Jan-Peter Ceglarek, polaris
@jan-peter:matrix.org
+4915772938111

Primary author: CEGLAREK, Jan-Peter

Session Classification: Room #1

Contribution ID: **22** Type: **lightning talk**

Manchester CubeSat Research Group

Saturday, 12 December 2020 16:26 (4 minutes)

University students have little to no experience in real life satellite design. The Manchester CubeSat Research Group helps educate and upskill space enthusiasts' students from different fields of study. Currently we are designing a bus intended for a mission range of 200 to 400 km, 50° to 90° inclination, circular orbit, 3U CubeSat with sufficient power and link budget to accommodate various optical payloads, including multispectral, TDI imagers. We believe our platform would be perfect for technology demonstrations or testing of on-orbit Machine Learning and Machine Vision technologies.

Primary authors: BARROS, Francisco (Manchester CubeSat Research Group); Mr BYROS, Antonis (Manchester CubeSat Research Group); Mr HUTCHINGS, Gregory (Manchester CubeSat Research Group)

Session Classification: Room #1

Contribution ID: **23** Type: **tutorial**

Python Implementation of CCSDS File Delivery Protocol (CFDP)

Saturday, 12 December 2020 16:45 (30 minutes)

CFDP is the CCSDS File Delivery Protocol, which is similar to FTP but designed for application in space communication, mainly spacecraft to ground communication and vice-versa. CFDP is a CCSDS standard and has been published many years ago but has not yet been used widely in space. Upcoming missions, like ESA's Euclid and JUICE missions will make heavy use of CFDP.

Also, CFDP plays a crucial role for the concept of "file-based operations", an operations paradigm that is expected to largely complement the current telecommand based operations.

Over the last several months, we have developed a Python implementation of the CFDP protocol. In this tutorial we will show you how to set up two CFDP entities on your computer to send files back and forth using the CFDP protocol.

For the near future, a Class 2 implementation is being worked on. Also, a porting of this Python module to MicroPython is foreseen.

Repositories:
https://gitlab.com/librecube/prototypes/python-cfdp
https://gitlab.com/librecube/prototypes/python-cfdp-gui

Primary author: Mr SCHOLZ, Artur (LibreCube Initiative)

Session Classification: Room #2

Contribution ID: **24** Type: **discussion**

Standardization of CubeSat PC/104 Connector

Saturday, 12 December 2020 17:15 (45 minutes)

The so-called PC/104 connector is present at almost any CubeSat board. It derives from the PC/104 standard, which uses a stacked configuration of boards that use this connector to route power and data through dedicated pins. There is, as the name suggests, a total number of 104 pins that can be used.

This connector was likely first used by Pumpkin, one of the earliest CubeSat vendors, but it was never formally specified or standardized. That resulted in a big variety of implementations that we have today. For example, some have put their data bus to those pins that users use for power routing.

We don't intend to force a standard uponthe CubeSat community, but instead, our intention is to ensure that open source CubeSat boards (such as Libre Space Foundation and LibreCube boards) remain compatibly to each other.

The objectives of this discussion are:
- Gather examples from heritage and common practice regarding the pin usage of PC/104 connector
- Draft a pin assignment that defines mandatory pins, optional pins, and user defined pins
- Seek for harmonization with the PocketQube pin assignments

Primary authors: Mr SCHOLZ, Artur (LibreCube Initiative); Mr STARCIK, Milenko (VisionSpace Technologies GmbH)

Session Classification: Room #2

Contribution ID: **25** Type: **talk**

Library Space Technology Network

Sunday, 13 December 2020 15:30 (20 minutes)

Since May 2019, librarians at the Center for Astrophysics together with staff from the Libre Space Foundation have been piloting a new initiative called the Library Space Technology Network (LSTN). The aim of LSTN is to broaden access to space technologies through a network of public libraries who have agreed to install SatNOGS-powered ground stations on their roofs, and engage their communities in understanding and using the ground stations. In this talk, we will give a brief update on our progress so far, which includes: selecting the libraries for the pilot, developing a ground station kit, creating educational materials, and starting to install the ground stations at partner libraries.

Primary author: CARVER, Nico (Center for Astrophysics | Harvard & Smithsonian)

Session Classification: Room #1

Contribution ID: **26** Type: **talk**

MetaSat: Metadata for SmallSat Missions

Sunday, 13 December 2020 14:10 (20 minutes)

The variety of SmallSat missions range from from high school science education programs to interplanetary communication support, and the ways people document their missions are as diverse as the missions themselves. To improve information sharing between missions and across systems, librarians at the Center for Astrophysics together with staff from the Libre Space Foundation have worked to develop MetaSat, a metadata toolkit for SmallSat missions. MetaSat currently has three primary components: a vocabulary of concepts, a set of metadata crosswalks, and a collection of JSON-LD example files. This presentation will update the OSCW community on MetaSat's development and implementation by SatNOGS, as well as where we hope to go from here. Attendees can find out more about MetaSat by visiting https://schema.space/.

Primary authors: WILLIAMS, Allie (Harvard-Smithsonian Center for Astrophysics); BOUQUIN, Daina (Harvard-Smithsonian Center for Astrophysics)

Session Classification: Room #2

Contribution ID: **27** Type: **lightning talk**

Mechanical design for space applications using open source tools

Sunday, 13 December 2020 16:22 (4 minutes)

During 2019 and the longest year ever recorded, Libre Space Foundation designed, built and delivered for launch 2 picosatellites named Qubik 1 and 2, and a picosatellite deployer named PICOBUS. Is open source CAD/FEA/CAM software mature enough to give answers in real world problems? This is the story behind the mechanical design, FEA analysis and construction of space hardware while chasing deadlines and constant challenges.

Spoiler: Everything ready to fly!

Primary author: PAPAMATTHAIOU, Manthos

Session Classification: Room #1

Contribution ID: **28** Type: **talk**

A Dual-Redundant CubeSat Flight Computer Based on Raspberry Pi

Sunday, 13 December 2020 15:10 (20 minutes)

The DualPi Flight Computer is a low-cost CubeSat on-board computer design based around a redundant pair of Raspberry Pi Compute Module 3 devices, for fault-tolerant operation in warm-to-hot redundant modes. This OBC design is the outcome of a long-running project developed at the Surrey Space Centre by undergraduate Electronic Engineering students from the University of Surrey. The design files have been made available for use by the open-source community.

The OBC was designed to the specifications of the AAReST (Autonomous Assembly of a Reconfigurable Space Telescope) mission's Payload Interface Computer. AAReST aims to demonstrate on-orbit assembly of large mirrors for space telescopes from smaller modular mirrors. The Payload Interface Computer's system interface capabilities include shared UART, I2C master and slave buses, a shared embedded USB Wi-Fi device (as an intersatellite link), and redundant support for Raspberry Pi Camera Modules.

This talk will present a development roadmap of the OBC, including planned future development, followed by a top-down discussion of the system design and concept of operations. This will include a functional-level technical overview of the key subsystems required to support the Raspberry Pi Compute Modules on the board and provide heightened fault-tolerance. A key area of interest is the embedded monitoring system, built around an MSP430 FRAM microcontroller; this detects critical errors within the main system through continuous state-of-health monitoring and responds to protect hardware from damaging faults such radiation-induced latch-up.

Primary author: PIRIE, Alex (University of Surrey)

Co-author: Dr BRIDGES, Christopher (University of Surrey)

Session Classification: Room #1

Contribution ID: **29** Type: **lightning talk**

Design, Fabrication, and Measurement of On-Board UHF Turnstile Antennas with Optimized Radiation Patterns

Saturday, 12 December 2020 16:18 (4 minutes)

In this study, on-board UHF turnstile antenna systems for CubeSat applications are designed in a simulation and optimization environment in order to obtain different radiation requirements. A turnstile antenna is a configuration with four monopoles that are placed circularly with successive 90° phase differences between them. The preference of turnstile structures in CubeSat systems is due to their inherent circular polarization and high gain when compared to monopole and dipole antennas, which are typically linearly polarized and omnidirectional. Instead of defining numerical specifications to conceptualize a design, the performance limits of turnstile antennas are first observed via optimization sweeps by studying the orientations of antenna arms in simulation environment. In order to feed antenna arms by successive 90° phase differences between them, a four-way quadrature splitter circuit is designed and fabricated. Antenna pattern measurements are conducted in an anechoic chamber by connecting monopole antenna arms to the fabricated power splitter circuit, while the arms are oriented in the optimized directions and placed on a generic 1U CubeSat platform. Integration of the designed antennas into real CubeSat systems with deployment mechanisms is under investigation.

Primary authors: DOLAPÇI, Türker (Middle East Technical University, Turkish Amateur Satellite Technologies Organization); ŞAHIN, M. Barış (Middle East Technical University); Dr ERGÜL, Özgür (Middle East Technical University)

Session Classification: Room #1

Contribution ID: **30** Type: **talk**

OpenSatCom: Open Source development methodology models for satellite communications

Saturday, 12 December 2020 14:55 (20 minutes)

OpenSatCom is a European Space Agency activity implemented by Libre Space Foundation in collaboration with Inno3 aiming to assess the relevance of open-source methodology models for the satellite communications sector, where such models can provide benefits and how can the European Space Agency contribute to assist relevant existing or new open-source projects.

A major output of this activity is the "Open Source Development Models for Satellite Communications" report.

This talk will discuss the findings of this report, the relevant projects identified, and the open-source sub-activities implemented as part of OpenSatCom.

Primary authors: KOSMAS, Eleftherios (Libre Space Foundation); PAPAMATTHEOU, Manthos (Libre Space Foundation); PAPADEAS, Pierros (Libre Space Foundation)

Session Classification: Room #1

Contribution ID: **31** Type: **talk**

An Open Source Implementation of the ECSS PUS Services in Rust

Saturday, 12 December 2020 14:15 (20 minutes)

We will present an ongoing project of software development for ECSS PUS-C services in the new robust and fast programming language Rust[1]. Vision Space Technologies[2] started this open-source project called Prust[3]. The standard ECSS-E-70-41C[4] is a packet utilization standard (PUS), defining the packet structure and a set of standard services. Its recent iteration PUS-C was published in 2016 and it's relatively new. PUS is the standard of choice by ESA, and only one spacecraft operated by ESA doesn't use PUS (OPS-SAT). A space project can choose to implement only those standard PUS services which suit its specific mission characteristics and requirements. Also since there can be additional requirements depending on the mission PUS also includes some guidelines and rules for how these extensions should be implemented. It is clear that implementations of PUS needs to be robust and safe given the usage of these softwares, also it should be maintenance-cost effective since requirements are very dynamic. We will present how we tackled these requirements in Prust project.

Prust project aims to create software that is reliable, fast, and encapsulated in a way that only the desired functionalities are dynamically loaded to the spacecraft. This is why we chose to implement the software in Rust. C remained the only alternative for a long time because it was faster than other programming languages. The speed of C came from not checking memory safety at runtime. But with its features, Rust offered a cost-free way of ensuring memory safety. It also offered usage of High-Level programming features with System-Level performance. This paper[5] says with the adoption of Rust these mechanisms will become commoditized. Prust was used in VST104 device which is another project of Vision Space Technologies.

Prust offers potential ease of abstraction while being fast and robust. Two complete PUS services and some sub-services are implemented already. Implemented services are sufficient for controlling and monitoring spacecraft peripherals. With the convenience of Rust, well-written error propagation and error-handling features (Similar to Exceptions in higher-level languages) are implemented without cost. Which is handled poorly in C in terms of maintenance cost and safety because of the nature of the language itself.

We will present why and how Prust is an important project because it uses a new programming language which can be the future of embedded programming we will also show the following implemented PUS services;

1. **Function Management (Service 8)**; One request (TC) and it executes a function defined by the user by giving the name of the function. Which offers a potential to change the usage of the spacecraft dynamically.
2. **Request Verification (Service 1)**; 9 response (TM) types which every one of them indicates different states for the sub-services send (for example failure,success etc..)
3. **Housekeeping (Service 3)**; Plenty of TM and TC packets in it. It helps to have reports of the peripherals connected to the device. Implemented partially to monitor peripheral values from the spacecraft.

Prust project is still active and we would like to contribute by showing you the details of this project.

References

[1]: https://www.rust-lang.org/

[2]: https://www.visionspace.com/

[3]: https://github.com/visionspacetec/Prust

[4]: ECSS-E-ST-70-41C – "Telemetry and telecommand packet utilization"
European Cooperation for Space Standardization
http://www.ecss.nl/, 15 April 2016

[5]: Abhiram Balasubramanian, Marek S. Baranowski, Anton Burtsev, Aurojit Panda, Zvonimir Rakamarić, and Leonid Ryzhyk.
System Programming in Rust: Beyond Safety. In Proceedings of the 16th Workshop on Hot Topics in Operating Systems (HotOS'17).
Association for Computing Machinery.
doi:10.1145/3102980.3103006, 2017

[6]: https://github.com/visionspacetec/VST104

Primary author: ÖZLEYEN, Selman

Co-authors: FEITEIRINHA, Jose (VisionSpace Technologies GmbH); GEIB, Filip (VisionSpace Technologies GmbH)

Session Classification: Room #2

Contribution ID: **32** Type: **talk**

What the hell was that?

Sunday, 13 December 2020 14:30 (20 minutes)

Sending a satellite to space is easier than ever before. Cubesats are relatively inexpensive but at the same time they are capable enough to track ships, watch for earthquakes, or even observe exoplanets. "Ease of launch" does not mean "ease of operation", though; the satellites send a wealth of telemetry back to earth, and turning that flood of information into action is difficult. This difficulty only increases as fleets scale in size, from one-off research projects to hundreds of satellites providing commercial services.

Since the beginning of 2018, the Open Source Polaris project has used Python's rich ecosystem to build a machine learning pipeline applicable to any mission. Now, since about summer 2020, it's doing so under the umbrella of the Libre Space Foundation.

Polaris analyzes telemetry for each satellite, automatically extracts dependencies among its components, and displays these in an interactive, browser-based 3D graph. Spacecraft operators can navigate their graph and understand relations between their telemetry. This not only gives operators another tool to monitor performance, diagnose problems, and predict satellite behaviour but also gives them a succinct, intuitive representation of all information being evaluated.

We have already presented Polaris in previous editions of OSCW, but this year we want to give an update to the Community about the most recent developments. In particular, we'd like to show how Polaris can detect anomalies in telemetry and how we use external parameters, such as space weather, to get better insight.

During last year, mainly thanks to Google Summer of Code 2020, we implemented an autoencoder based (deep learning) algorithm which creates concise representations of the telemetry data for each timestamp. By finding the vector distance between "neural kernels" in consecutive timestamps, we get a number which tells us how far an esnemble of telemetry has deviated with time. By setting a threshold for these numbers, alerts can be triggered whenever an anomaly is detected (i.e. a high value). The reason why this works well is because as the model learns the best and minimum representation for the telemetry data, it also learns how a change in one data point is likely to affect the rest of the data points, allowing us to automatically amplify or subdue the important and not so important/routine events.

From the space weather side, we added support for proton, electron, magnetic, gamma and solar events, with data sourced from swpc.noaa.gov. This addition is especially useful as the effect of space weather on the cubesat hardware (reflected by the telemetry) can easily be observed.

We want to open the discussion with an update to the project as described above and then open the floor for the audience to give us feedback, request features and brainstorm about useful ways to integrate machine learning analysis into spacecraft operations.

Primary authors: BOUMGHAR, Red (Libre Space Foundation); CRESPO, Xabier (LSF Core Contributor)

Co-authors: VENKATESWARAN, Adithya (Polaris); CEGLAREK, Jan-Peter; FLAWINNE, Julien; CROISSANT, Kevin (Ohio University)

Session Classification: Room #1

Contribution ID: **33** Type: **lightning talk**

Pico Fluid Dynamic Actuators in CubeSats

Sunday, 13 December 2020 16:26 (4 minutes)

As CubeSat capabilities continue to expand as the number of missions increase each year, the Attitude Determination and Control or (ADCS) subsystem is a major hurdle for CubeSat developers to undertake. The ADCS subsystem commonly contains actuators that are used to change the attitude of a CubeSat by means of reaction wheels. This proposal aims to present the use of alternate novel pico Fluid Dynamic Actuators or (pFDAs) to provide 3-axis attitude control for a 1U CubeSat. In contrast to bulky reaction wheels pFDAs offer an increase in payload volume optimization and can be configured in a 3 or 4 actuator variant to provide system redundancy. Additionally pFDAs feature no moving parts and can provide faster or more immediate torque. The use of pFDAs can now enable CubeSats with novel mission scenarios such as Artificial Swath Increase (ASI) and Single-Spacecraft Stereo-Imaging (SSSI). Ongoing research and development of pFDAs will feature testing in suborbital test flights onboard the Technische Universität Berlin Picosatellite EXperiment 7 (TUPEX7). With the use of low cost manufacturing technologies like 3d printing, commercial off the shelf electronics and open source communication protocols such as SPI, I2C, CAN pFDAs can disrupt trade off analysis and unlock new CubeSat potential.

Primary authors: GARCIA CACHO LASCURAIN, Diego; Dr GRAU, Sebastian; Mr OSPINA, Sebastian; Mr DIEZ, Jose

Session Classification: Room #1

Contribution ID: **34** Type: **talk**

Goodbye TLEs, hello OMMs!

Sunday, 13 December 2020 14:10 (20 minutes)

Tired of squinting at punched cards to find out the orbit of your satellite? Fear no more! Two-Line Element sets, most widely known as TLEs, are coming to and end, and a not-so-new format, OMMs, is set to have a huge relevance in the coming future. But what tools do you have at your disposal to deal with this format? And is it really a format, or a family of them? As part of the OpenSatCom initiative spearheaded by the Libre Space Foundation we studied the status of the open source ecosystem regarding OMM support, and in this talk we will offer a summary of our findings.

Primary author: Mr CANO RODRÍGUEZ, Juan Luis

Session Classification: Room #1

Contribution ID: **35** Type: **tutorial**

How to contribute to open source

Saturday, 12 December 2020 16:45 (1 hour)

In this tutorial we will give a brief introduction of how most free & open source projects are developed these days, we will summarize the differences between GitLab and GitHub (arguably the two most popular platforms to host source code nowadays) and make a git walk-through to make our first contribution to a (dummy) project.

Attendees will not need to have anything installed, just a reliable Internet connection and a GitLab account.

Primary author: Mr CANO RODRÍGUEZ, Juan Luis

Session Classification: Room #1

Contribution ID: **36** Type: **lightning talk**

SunrIde a Student Venture to Democratise Access to Space

Saturday, 12 December 2020 16:30 (4 minutes)

SunrIde is the Sheffield University Nova Rocket Innovative Design Engineering project. Fundamentally we aim to allow students and in future the broader public to experience and access near space environments and gain the skills and trainings necessary to pave the way for a revival of the UK space industry. Established in 2017 SunrIde engineers have been working tirelessly to expand and diversify our impact in the UK Space sector improving the practical learning experience for hundreds of Students along the way. In 2020 the Project has expanded its teaching programm to encompass over 200 students and is leading the charge in building a University network of Space inspired Students with the mission of going beyond the Karmen line. Through this platform we wish to convey our journey from 2017 to present, our vision for the future and our practical and potential impact on the UK space Industry over the coming decade.

Primary author: DENT, Alec (SunrIde)

Session Classification: Room #1

Contribution ID: **37** Type: **talk**

Doppler Based Orbit Determination with Optimization

Sunday, 13 December 2020 15:10 (20 minutes)

Space Situational Awareness (SSA) is becoming more and more important as the number of satellites is increasing. The body of knowledge for SSA relies on the tracking data to determine the orbit of artificial objects in Earth orbit. Determining orbit is basically obtaining the state vector, that is the position and velocity vectors, or equivalently obtaining the classical orbital elements at a specific time. The state vector uniquely determines the orbit of a satellite, and the accuracy of this orbit determination is important to pursue a successful space mission. There are different kinds of orbit determination techniques which relies on different kinds of tracking data. In this study, the Doppler data of a single pass are used to determine the orbit of a satellite.

Guier and Weiffenbach discovered the potential of the Doppler effect to determine orbit of a satellite. It was the very beginning of the Space Era with the launch of Sputnik in 1957. They published their first paper, Theoretical Analysis of Doppler Radio Signals from Earth Satellite, on Nature in 1958. They searched for orbits of the Sputnik 1 and Explorer 1 in the paper.

Doppler shift is a fundamental phenomenon for the waves. It can be summarized as the change in the frequency due to the relative motion of the source and the observer. The change in the frequency can be calculated as $\Delta f = \frac{\Delta v}{c} f_0$ where c is the speed of light, Δv is the difference between the velocity of the source and the receiver, and lastly f_0 is the transmitted frequency. Thus, Δv gives the rate of change of the distance between the source and the observer, and it can be represented as $\dot{\rho} = \vec{v} \cdot \left(\frac{\vec{\rho}}{\rho}\right)$ where $\vec{\rho}$ is the relative position vector of the source with respect to the receiver. This means that Δf equals 0 for the case when the $\vec{v}$ and $\vec{\rho}$ vectors are orthogonal. This happens at the maximum elevation of the satellite with respect to the observer which is the configuration corresponding to the closest distance between the satellite and the observer.

In this study, the Doppler data are obtained from the observations and also generated from the TLE data. To generate artificial Doppler data from TLE, a two-body propagator is used to simplify calculations. Then, the Doppler data are used to determine the orbit. We aimed to determine the orbit by using minimum information of the observed satellite. Simply, we just "heard" the satellite. Under this uninformed situation, it is convenient to apply optimization techniques. For this application, there is a need for having some constraints on the parameter space of the possible orbit. Mainly, the satellite is assumed to be in the low Earth orbit. Thus, the altitude is constrained to be between 200 km to 2000 km. This conraint provides the domain of the semi-major axis and the upper bound of the eccentricity. Actually, these assumptions decreased the search space of the optimization and made the application affordable in terms of computation time.

To have the preliminary orbit determination from the Doppler data, the optimization technique that we used is the particle swarm optimization (PSO). After having the preliminary orbit determination with PSO, a gradient based optimization is further used to increase the accuracy of results. Then, differential correction is also used to further increase the accuracy.

We applied this approach for both the generated artificial data and the real data. In the real data that we obtained from our ground station $\#510$ in the SatNOGS network, there was a difference between the received transmitter frequency and the specified transmitter frequency of the satellite, and this fact decreased the accuracy of our results.

To conclude, we studied the Doppler data based orbit determination for uniformed situations of the satellite orbit. For the cases of near circular orbits, the results become more accurate. In addition, if the exact transmitted frequency info is known, it is possible to obtain fairly accurate orbit determination results.

Primary author: Mr ERTÜRK, Mehmet Fatih (Istanbul Technical University & University of Turkish Aeronautical Association)

Session Classification: Room #2

Contribution ID: **40** Type: **talk**

AcubeSAT, a 3U CubeSat to probe gene expression in LEO: technical & programmatic updates

Saturday, 12 December 2020 15:35 (20 minutes)

AcubeSAT is a 3U CubeSat designed to perform a biological experiment, probing the dynamic regulation of gene expression of eukaryotic cells in Low Earth Orbit (LEO), utilizing advances in Synthetic Biology and MEMS (micro-electro-mechanical systems). Besides potential experimental readout, our goal is to demonstrate that the payload design can be utilized for a variety of applications, therefore rendering multiplexed cell analysis in-orbit considerably more approachable.

Over the past year, after having been accepted in ESA's Fly Your Satellite! Programme, the AcubeSAT team has been working hard to improve the design also presented in OSCW 2019. The culmination of our efforts was the submission of the CDR, a 1100+ page collection of documents detailing mission and spacecraft design. With contributing to open science, open access and open source still being our main drive, we pushed to share additional information, including milestone-documents that, while part of every space mission, can be scarcely found online, if at all.

Our presentation has a two-fold goal. By briefly recounting our progress, we aim to spark a conversation regarding some aspects of space-mission design where the open-source community is still severely unrepresented, and ways to change this overall situation for the better. Additionally, we hope to share some lessons learned, aiming to bridge the gap between the legacy aerospace development methodologies and the agile approach preferred for low-cost access to space.

Primary authors: RETSELIS, Anastasios-Faidon (Aristotle University of Thessaloniki); OUSOULTZOGLOU, Orestis; Mr KANAVOURAS, Konstantinos (Aristotle University of Thessaloniki); Mr ARAMPATZIS, Asterios (Aristotle Space and Aeronautics Team)

Session Classification: Room #1