OPS-SAT in orbit - a technical rundown of this open experimentation platform

Dominik Marszk, David Evans, Maximilian Henkel, Otto Koudelka, Manuel Kubicka, Mario Merri, Tom Mladenov, Kevin Otto, Mehran Sarkarati

DASIA 2021 - 21/09/2021

About me

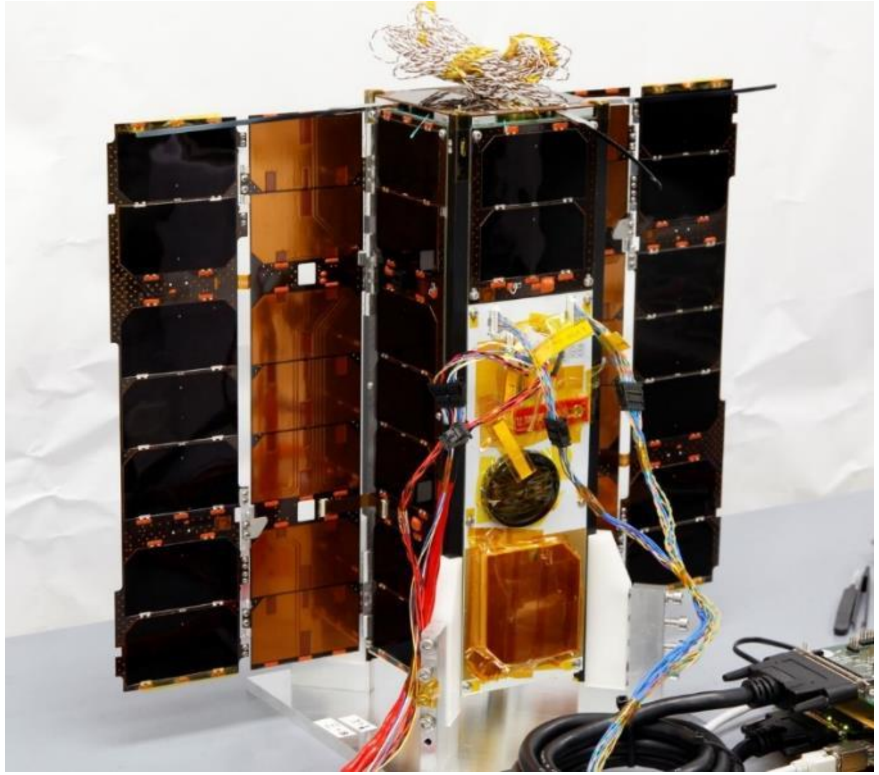
- Dominik Marszk
- Born and raised in Gdańsk, Poland
- Microelectronics engineer by education
- Software engineer by trade
- Data Systems Engineer in ESA/ESOC
- Started in ESA as a Young Graduate Trainee 5 years ago
- Data Systems Manager of OPS-SAT

Presentation plan

- What is OPS-SAT
- Space Segment
- Ground Segment
- Operations during pandemic
- Data Systems
- Experimenter framework
- Current Mission Status

What is OPS-SAT

- 3U CubeSat (ESA's first)
- An open hardware/software innovation platform in Low Earth Orbit
- De-risk concepts before deploying them on larger satellites
- 175+ companies from 17 countries registered experiments
- 55 additional experiments via recent OSIP campaign
- Academia, start-ups and large corporations are looking to innovate on OPS-SAT
- Experiments range from telemetry/image compression algorithms to experimental IP-cores on the FPGA
- Owned and operated directly by ESA
- Launched 18th Dec 2019 from Kourou (VS23) in a 515km SSO together with CHEOPS and COSMO-SkyMed SG
- Break the „has never flown, therefore will never fly" cycle



Space Segment – System Overview

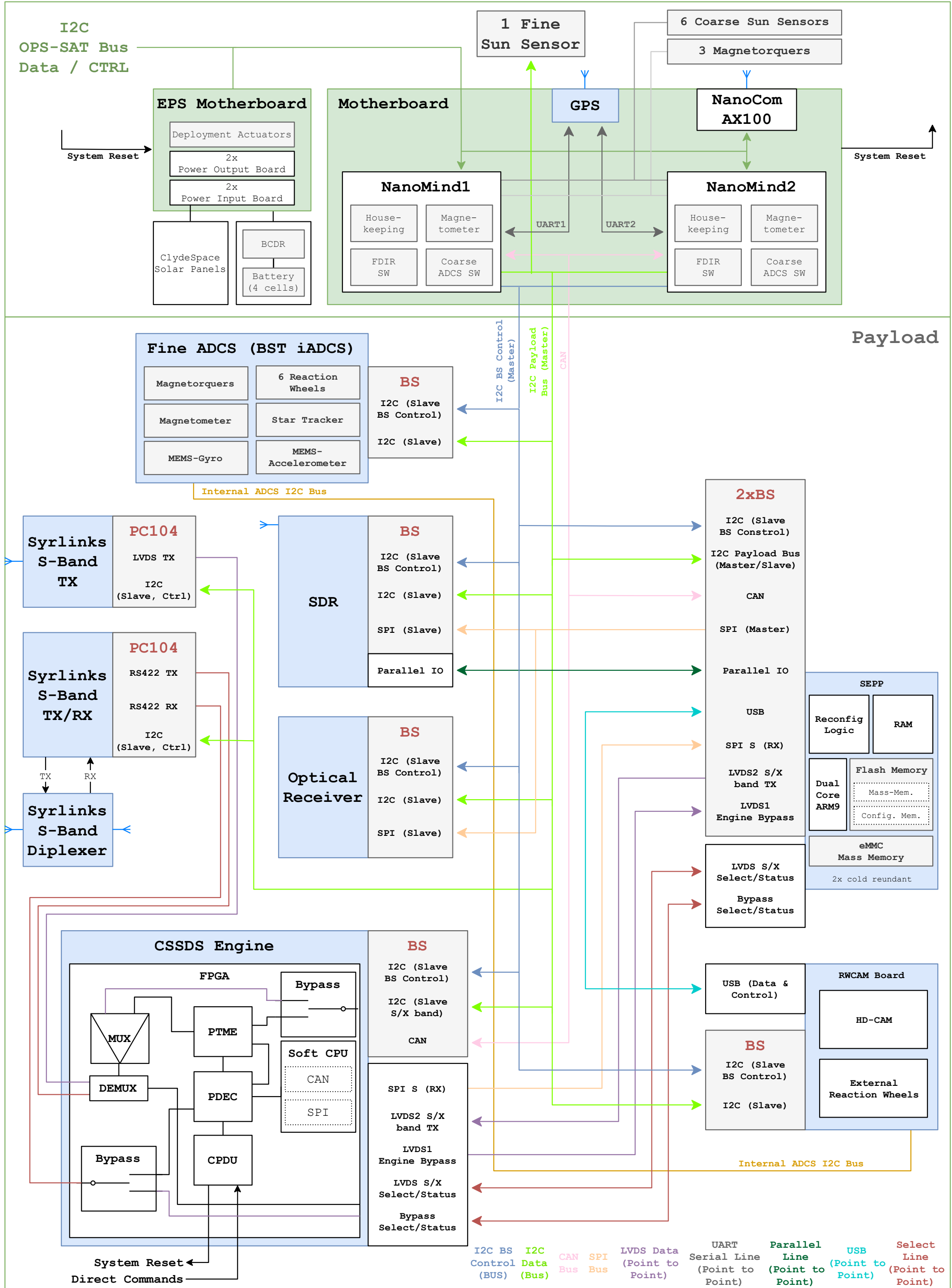
„Satellite in a satellite" design

COTS Satellite bus:

- Gomspace UHF AX100 radio + EPS/ACU
- Nanomind A3200 OBC (On-board computer, AVR32)
- GNSS receiver

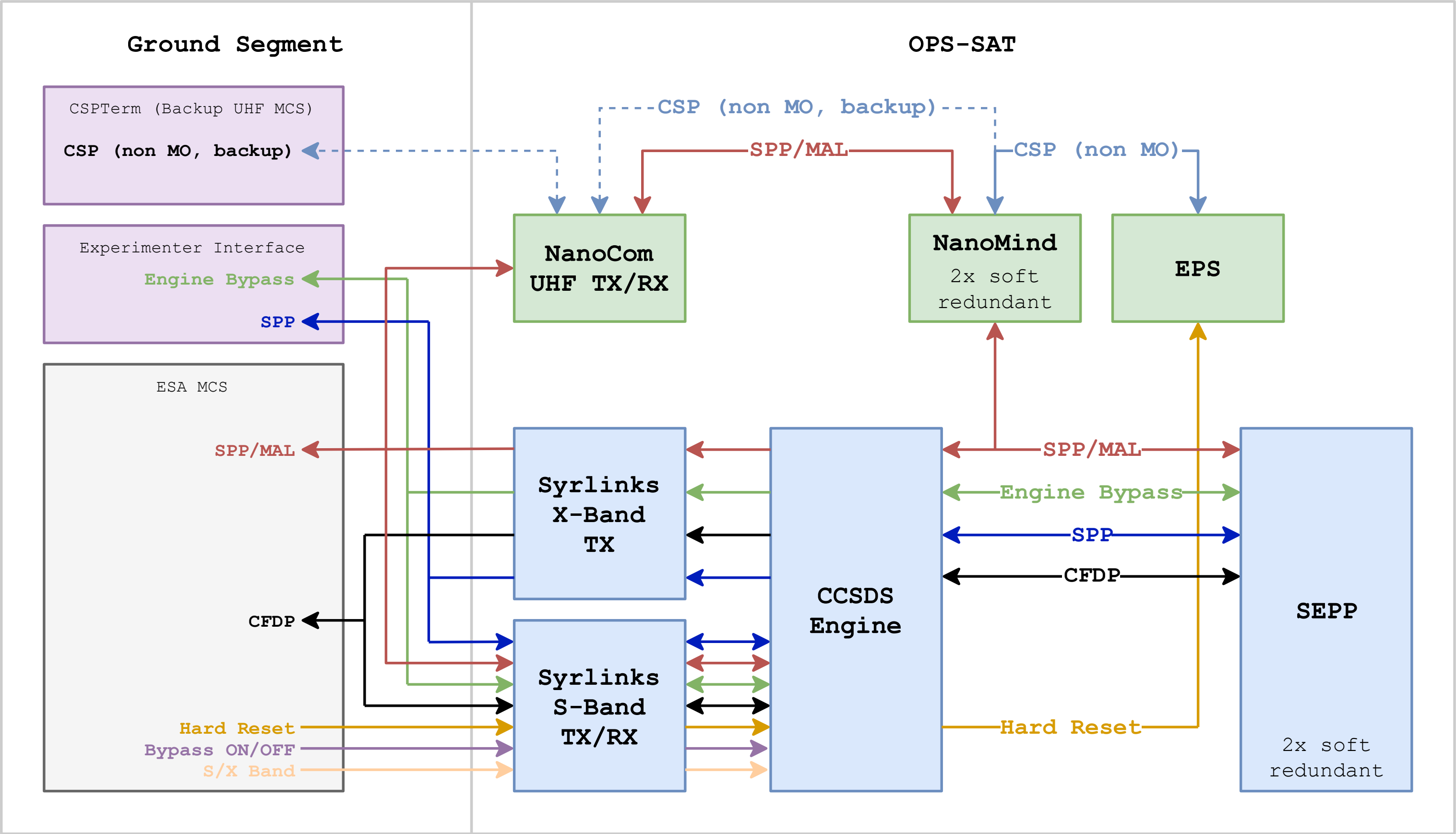
Experimental bus:

- S-band TM/TC transceiver
- X-band TM transmitter
- Software Defined Radio (LMS6002D)
- HD-camera (Nadir-facing)
- Optical receiver (data uplink via laser)
- Advanced iADCS (Attitude Determination & Control Sys.)
- 2x Cyclone V SoC (800MHz Dual Core ARM Cortex-A9 + FPGA fabric)



Space Segment – TM/TC Handling

- Core of the data handling in space is CCSDS Engine, multiplexing S/X-Band
- UHF used as contingency and for low level control of the COTS bus (9.6 kbps)
- Syrlinks S-band transceiver (256 kbps↑ / 1 Mbps↓)
- Syrlinks X-band transmitter (3-50 Mbps)
- Optional CCSDS Engine Bypass
  (still not utilised on daily basis)



Space Segment – SEPP

- SEPP - Satellite Experimental Processing Platform
- Core of the experimental platform
- Developed by TU Graz (mission prime)
- Running embedded Linux (built on Angstrom + Yocto 2.4.4 platform)
- Operated like a remote Linux machine (remote shell, package manager)
- Recently introduced TCP/IP connectivity over a space link
- A base software stack:
    - Java Runtime
    - Python 3.5
    - Payload API user-space libraries
    - NanoSat MO Framework – high level application framework

Ground Segment – SMILE lab & groundstations

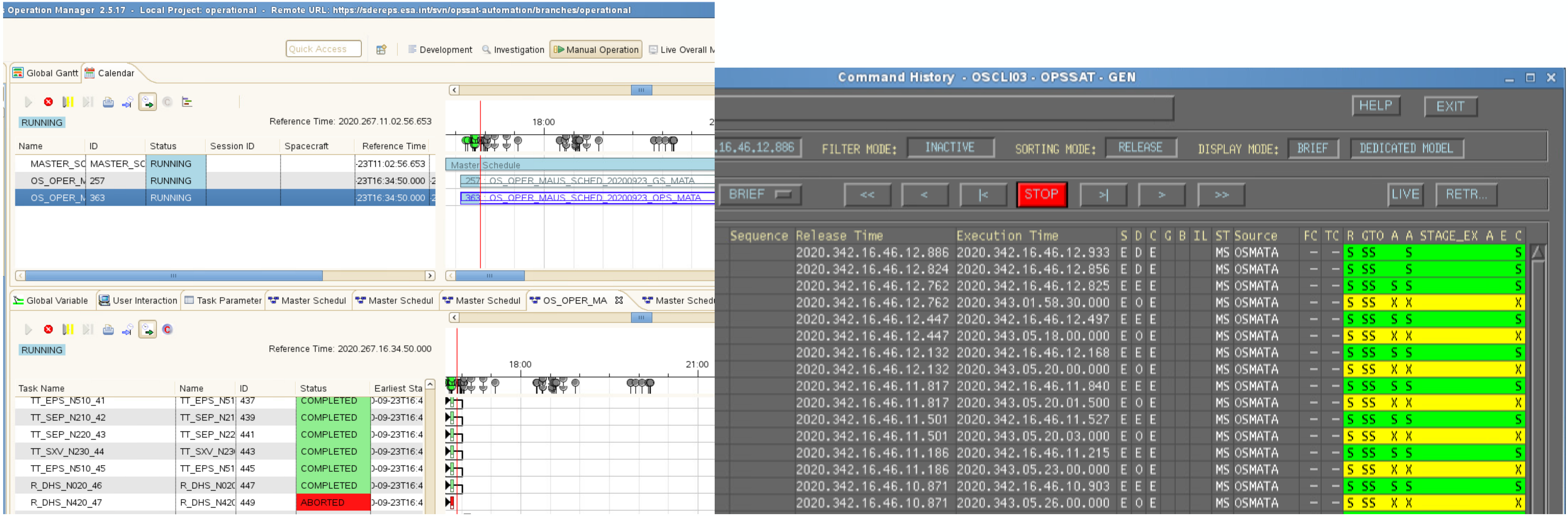
➢ IE-01 (UHF), Ireland

➢ ESOC-1 (S/X) and ESOC-1 (UHF), ESOC

➢ TUG-1 (UHF), Austria

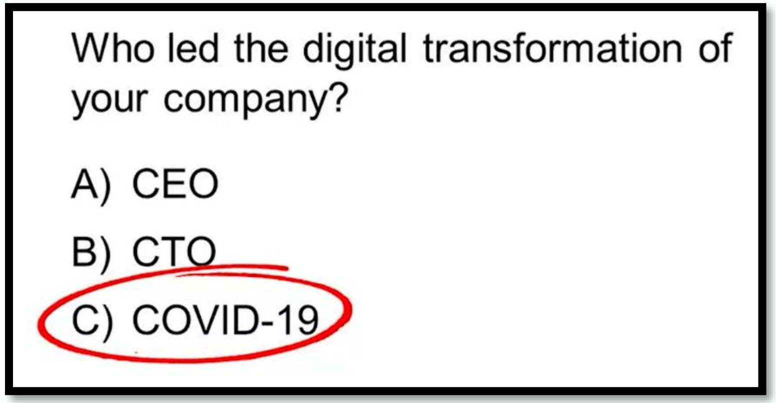
Ground Segment - MCS

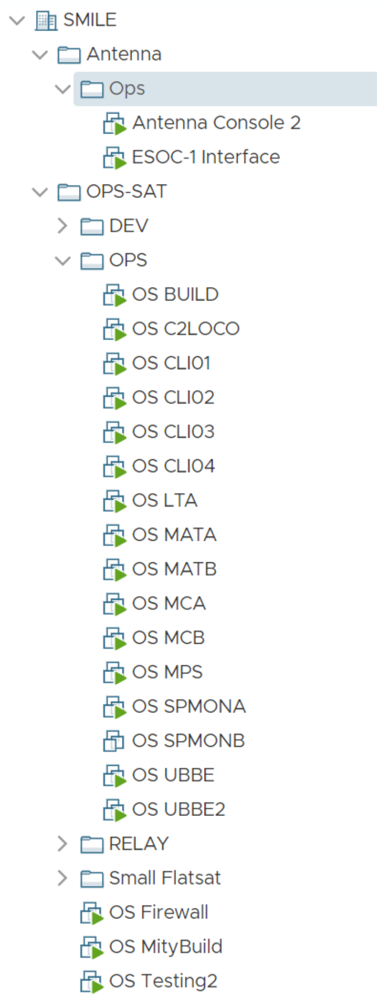
- SCOS + MATIS – standard for nowaday ESA missions (e.g. Gaia, EXM, BepiColombo)
- Procedures are triggered, uplinking 1 or more Time-Tagged (TT) commands
- Preprograming mission timeline for 24 or more hours



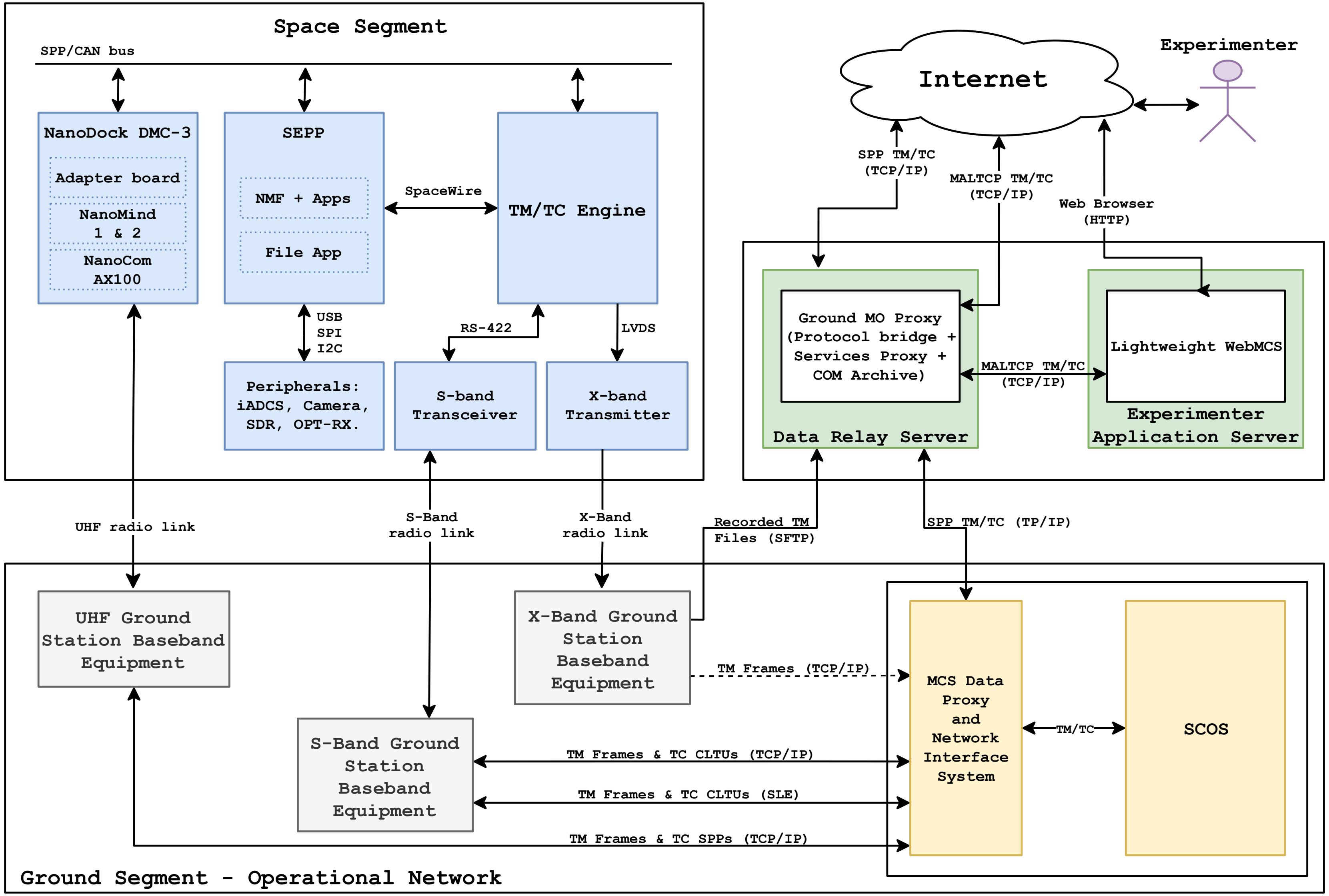
Operating a mission 99% remotely

- Heavy use of virtualisation - a number of VMWare hosts running our infrastructure – total around 40 VMs
- Move to teleworking was not very revolutional infrastructure-wise
- Hardware-contingency response is manageable
- DevOps concepts long before pandemic:
  - Mission-dedicated group on our internal GitLab – 88 projects
  - CI/CD ranging from runs of telemetry processing tools to builds of OBSW and MCS
  - Most projects go from merge of a patch to deployable, preliminarily validated artifacts within minutes
- Most of evening passes are manned while morning ones are unmanned
- Heavy use of procedural automation





Data systems & interfaces diagram



- Core of the data handling on ground is MCS Data Proxy, multiplexing:
    - UHF/S/X-Band on space-facing side
    - SCOS and Experimenter chain on ground-side

Experimenter Data Systems

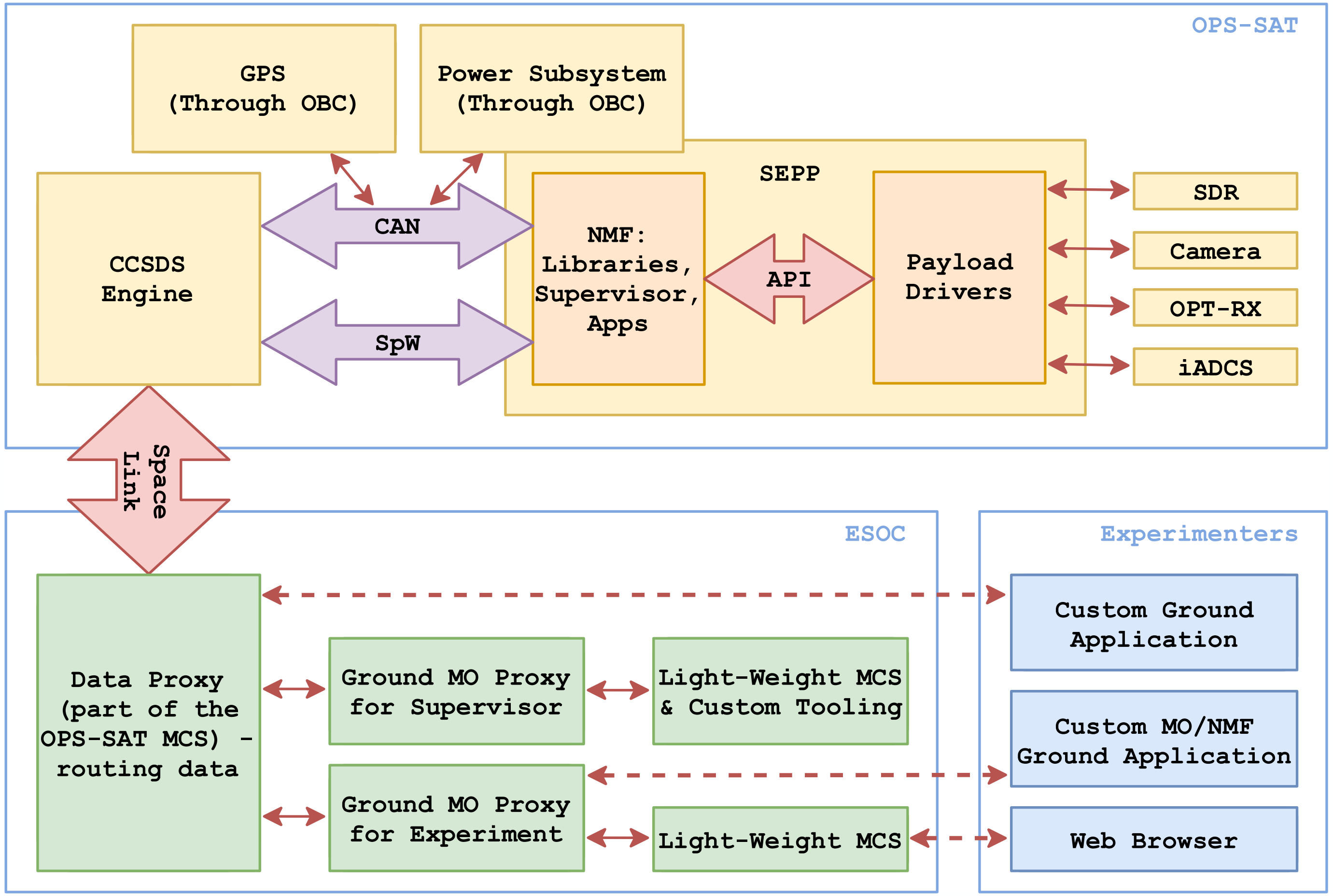
Variety of access interfaces both onboard and on the ground.

On-board:

- NanoSat MO Framework applications
- Native apps

Ground:

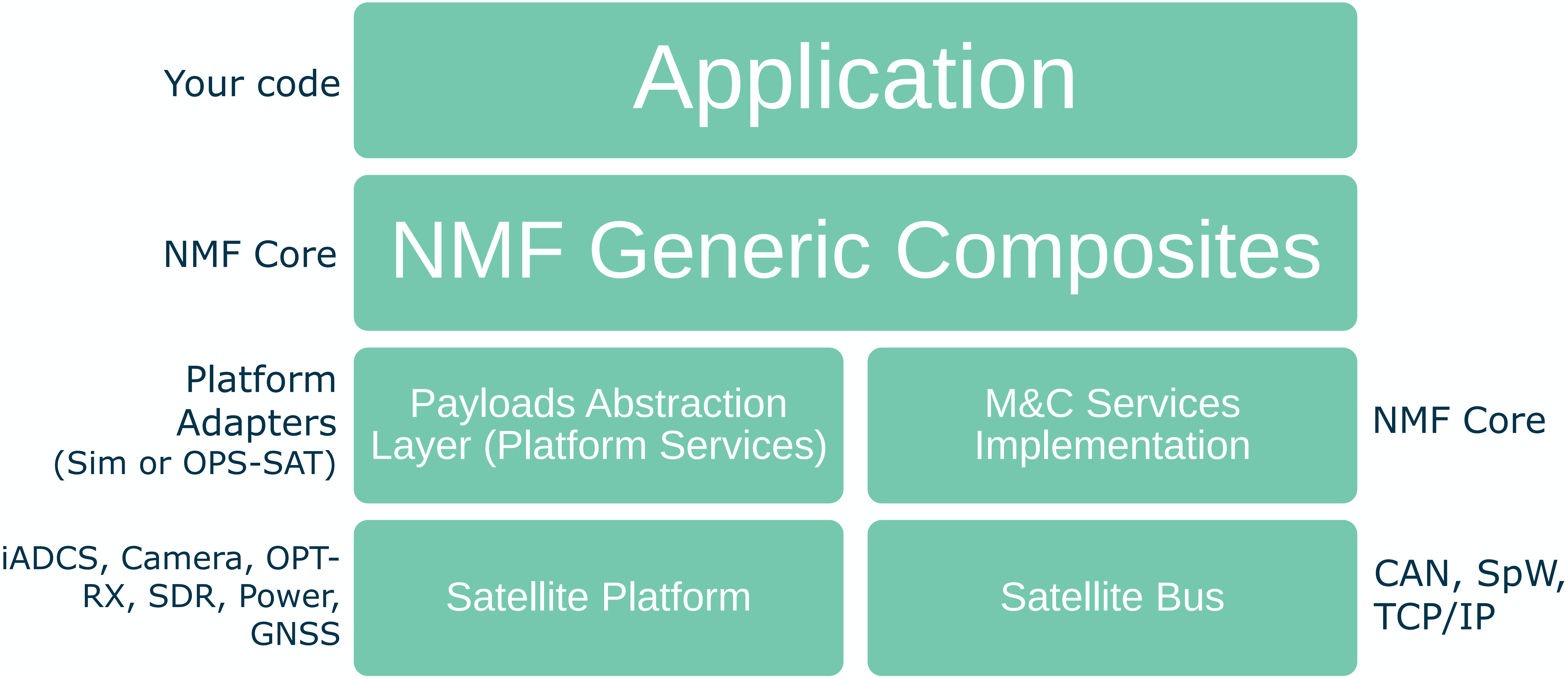
- Custom ground application
- Custom MO/NMF applications
- Web browser via Light-Weight MCS



Nanosat MO Framework – Capabilities

- Monitoring & Control
- Payloads Abstraction Layer
- Supervises applications through a management component (NMF Supervisor)
- Implements handling of basic signals required by the OPS-SAT system
- Space to ground replication (Ground MO Proxy)
- Offline mirroring of the M&C data on the ground

Nanosat MO Framework – Space app stack



NMF SDK – Simulator

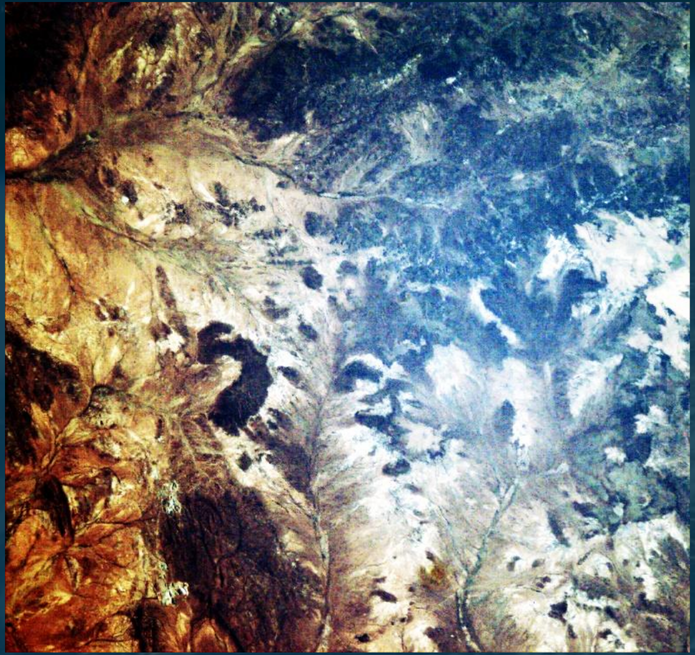
- Simulates core OPS-SAT payloads - GPS, iADCS, Camera
- Allows executing simulation scenarios
- Time control
- Portable implementation
- Hybrid mode
- Control GUI

Light-Weight Mission Control System

- Web interface hosted on ESOC premises
- High level consumer of M&C services
- Visualising the telemetry (Parameters and Alerts)
- Browsing history of the telemetry
- Telecommanding in an organised manner (Actions)
- Browsing history of the telecommands
- Available to experimenters as Docker containers
- Created as an R&D activity – still maintained

Current mission status

- Launched 18 December 2019
- LEOP concluded end of 2019
- Major part of commissioning finished Q1/Q2 2020
- Just extended until Q4 2022
- Hosted tens of experiments so far, and more in the pipeline
- Some experiments introduced into daily operations (e.g. FAPEC image compression)
- As experimenter complexity grows along with ageing hardware, the team is posed with new challenges every week



Thank you!

More pictures on our interactive map
https://ops-sat.io.esa.int/smartcam-map/

Dominik.Marszk@esa.int