The Third National Workshop on the SKA Project

Italian Role in SKA-Safe

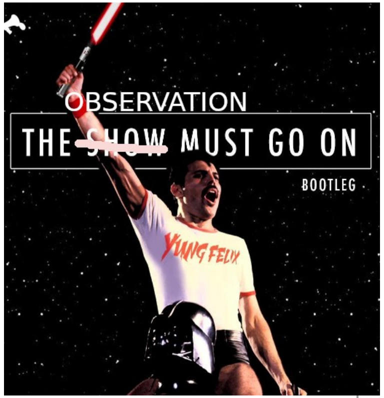
G.Comoretto, C.Baffa for the Italian **Cream Team**

04 Oct 2021

INAF ISTITUTO NAZIONALE DI ASTROFISICA NATIONAL INSTITUTE FOR ASTROPHYSICS

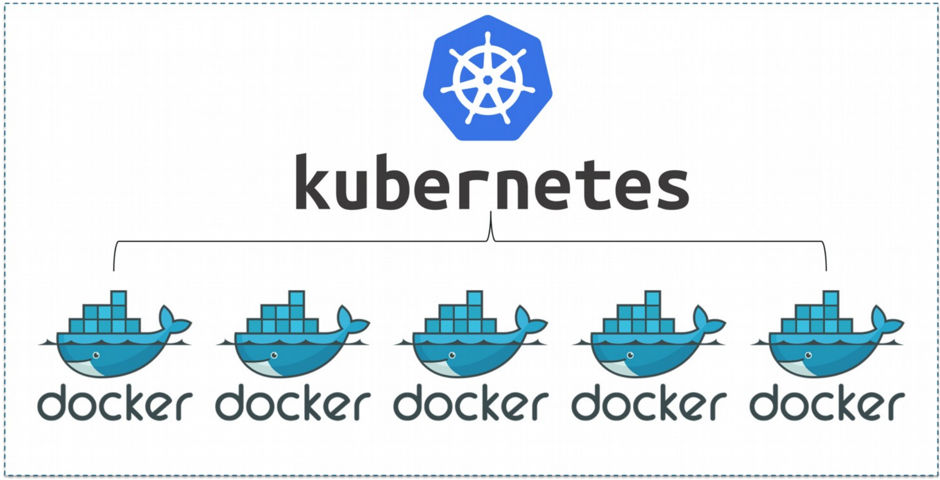
SKA software system - challenges

- Same software for two different telescopes
- Huge system
  - 100K elements. Faults will happen
  - 10^5-10^6 Flexible interconnections
  - Tolerant to local failures and configuration changes
- Flexibility
  - Automatic dynamic scheduling
  - Concurrent observations: subarrays (up to 16)
    - Completely independent
    - Resources (stations, correlator units) shared between subarrays
- Work organization:
  - scattered teams
  - Different cultures, time zones, expertises



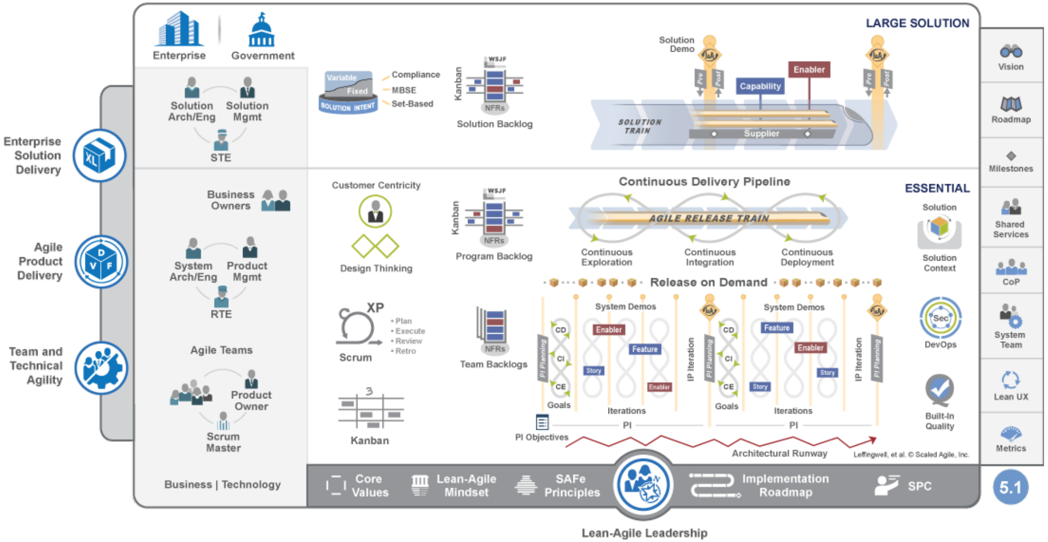
SKA software system - solutions

- Tango controls
  - Distributed control system for large installations
  - Widely supported
- Containers: Docker, kubernetes
  - Virtual environments, assembled from pre-compiled blocks
  - Easy to orchestrate, deploy, upgrade
  - Resilient to failures
  - Extra layer of complexity
- SKA base classes
  - Find commonalities in the system and program them once
  - Include generic features like hardware support, power on/off, simulation, health monitoring
  - Now at version 11, still evolving (too fast)
- Generic and powerful development and control tools
  - Emphasis on Graphic User Interfaces



Work organization: the SAFe environment

- Management architecture for large industrial projects
  - High overhead: 10-15% of the time spent planning
  - Small teams, self organized
  - Work partitioned hierarchically:
- Good
  - Splits things in manageable units
  - Forces to plan things
  - Eases collaboration
- Limits
  - Requires dedicated people or overhead is too high
  - Difficult to plan workload in a research environment
  - SKA SAFe is a bit “tamed” for a research environment



Safe structure

SKA Italian SW Areas

4 main topics:

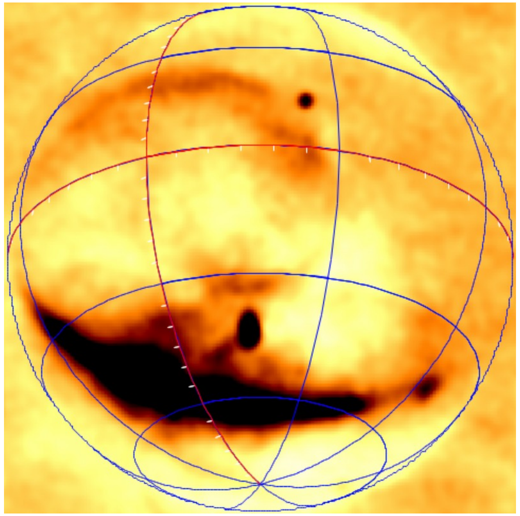
System infrastructure and services	M.DiCarlo
Low Frequency Aperture Array interface to Local Monitor and Control	G.Comoretto, S. Chiarugi, C. Belli
Control and monitoring of Central Signal Processing	E.Giani, G.Marotta, C.Baffa
Taranta Graphics User Interface engine	M. Canzari, V.Alberti

People in blue have also administrative tasks: PO (CB) and SM (VA)

LFAA Software & Firmware

LFAA: the “dishes” of SKA low. Stations of 256 antennas

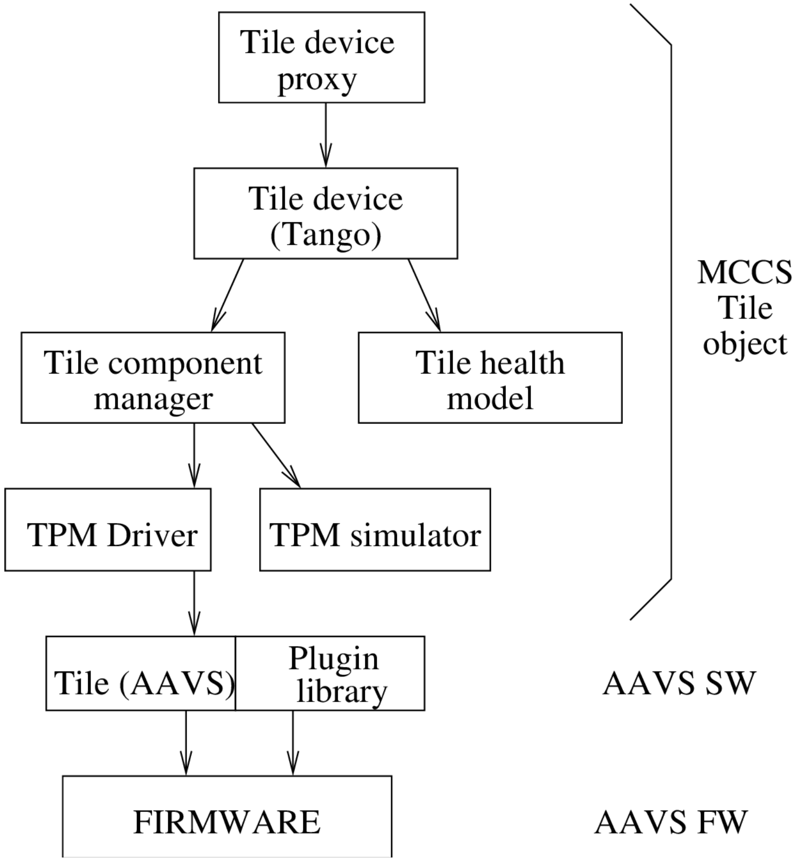
- From a (good) protype to the final thing
  - Standalone LFAA system
  - Good enough for all-sky mapping with single station
- Firmware
  - Firmware is managed like the software
  - 90% compete. Remaining 10% requires the other 90% of the effort
  - Improve code quality: Continuous integration, automatic unit testing
- Low level software:
  - Enough functionality to control a single station
  - Represent the lower layer of the SKA control software



INAF ISTITUTO NAZIONALE DI ASTROFISICA NATIONAL INSTITUTE FOR ASTROPHYSICS

LFAA Software

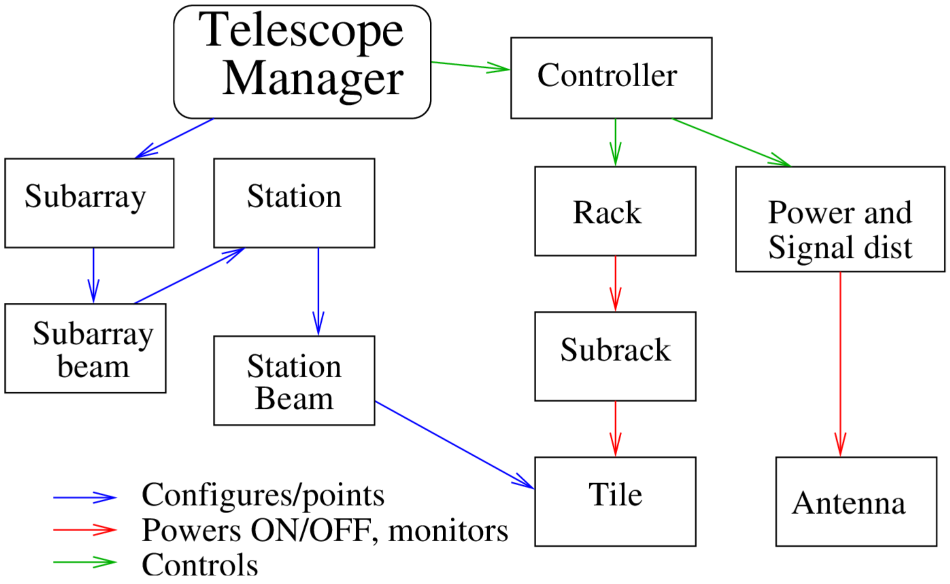
- MCCS: the local monitor and control for SKA-low stations
  - Configures and monitors the hardware
  - Based on Tango Controls, but lots of add-ons to standardize common functions
  - Each SKA device is composed of 5-8 interacting objects
- Possible to share stations between subarrays. Example configuration with 3 subarrays
  - Full array for high resolution HI mapping (150 MHz BW, at z=3-20)
  - Pulsar search (100 MHz bandwidth)
  - High sensitivity narrow band survey using multiple beams (8 beams x 6.25 MHz bandwidth)



MCCS Software

- Complex structure, must manage:
  - Independent beamforming (48 beams/station)
  - Allocation of beams to subarrays
  - Calibration per beam/station
  - Fault management
- Hardware elements:
  - 15,000 physical devices
  - 30,000 IP addresses
  - 1 Million control/monitor points
  - 150,000 Tango devices
- INAF: hardware element interface
  - Tile: signal processing platform for 16 antennas
  - Subrack: power and signal distribution for ½ station
  - Rack: Whole LFAA signal processing for 2 stations
  - PASD: Station RF and power electronics on the field

MCCS software structure



Cream Team Effort: Why?

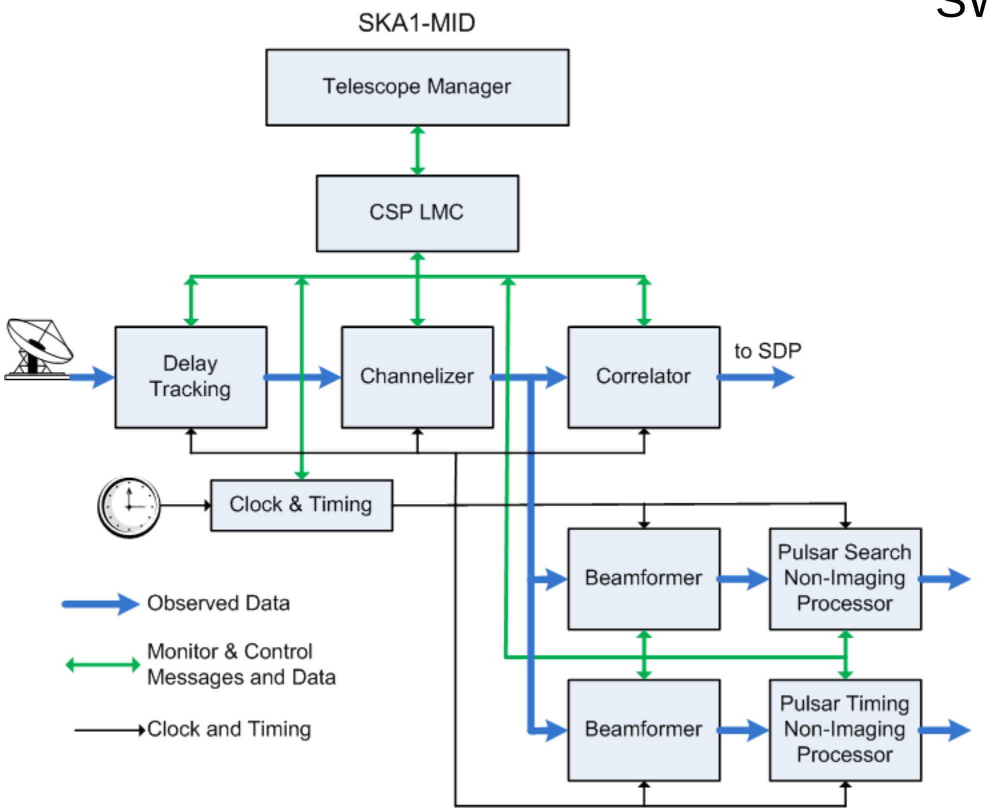
Two main areas of work:

- CSP control software:
  - Central Signal Processing: main data processing area.
  - Need to be managed in a fast, versatile and efficient way:
    - A wrong control design which let observe only a portion of time is equivalent to have a smaller telescope
  - Enable multiple possibilities as different and complex configuration and commensal observations.

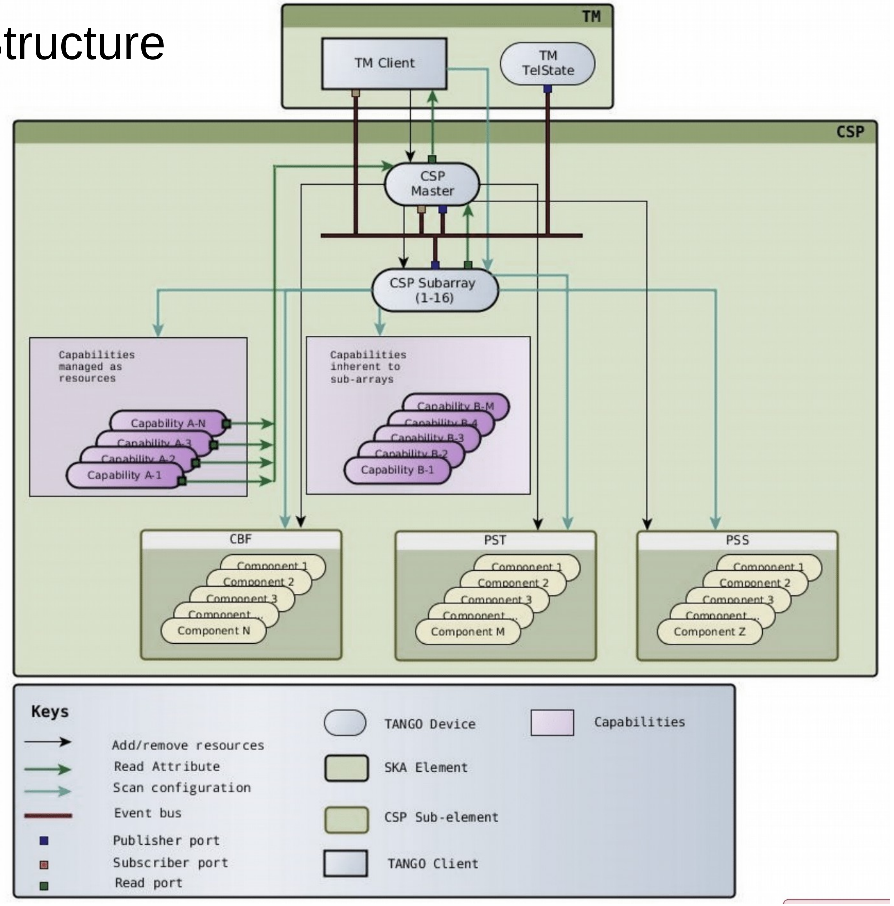
- Taranta GUI engines:
  - Started as engineering interface now evolving towards a general one.
  - Web only: no local installation required. Very handy but some complexity implied.

CSP Control and Monitor.

Data flow

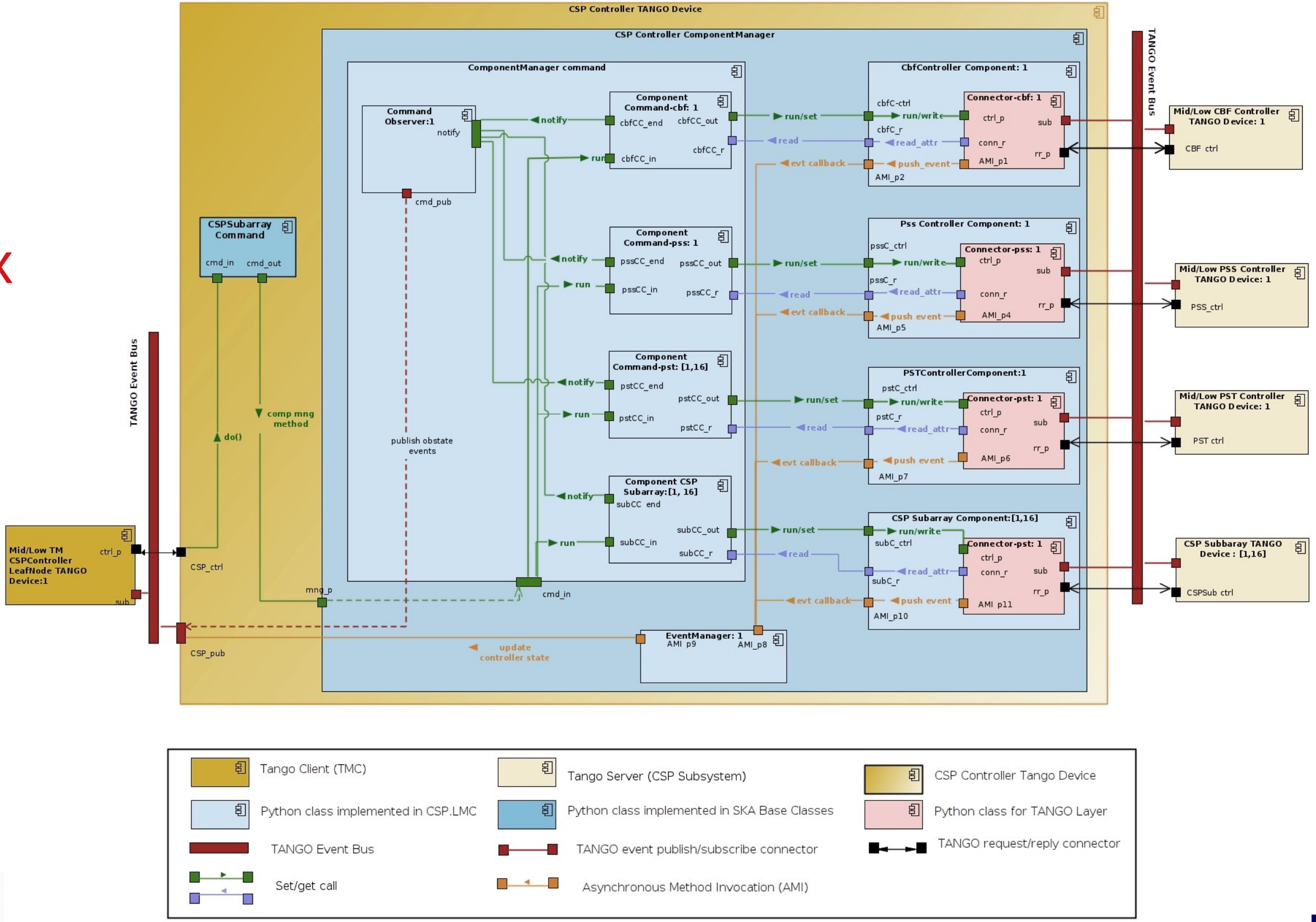


SW Structure

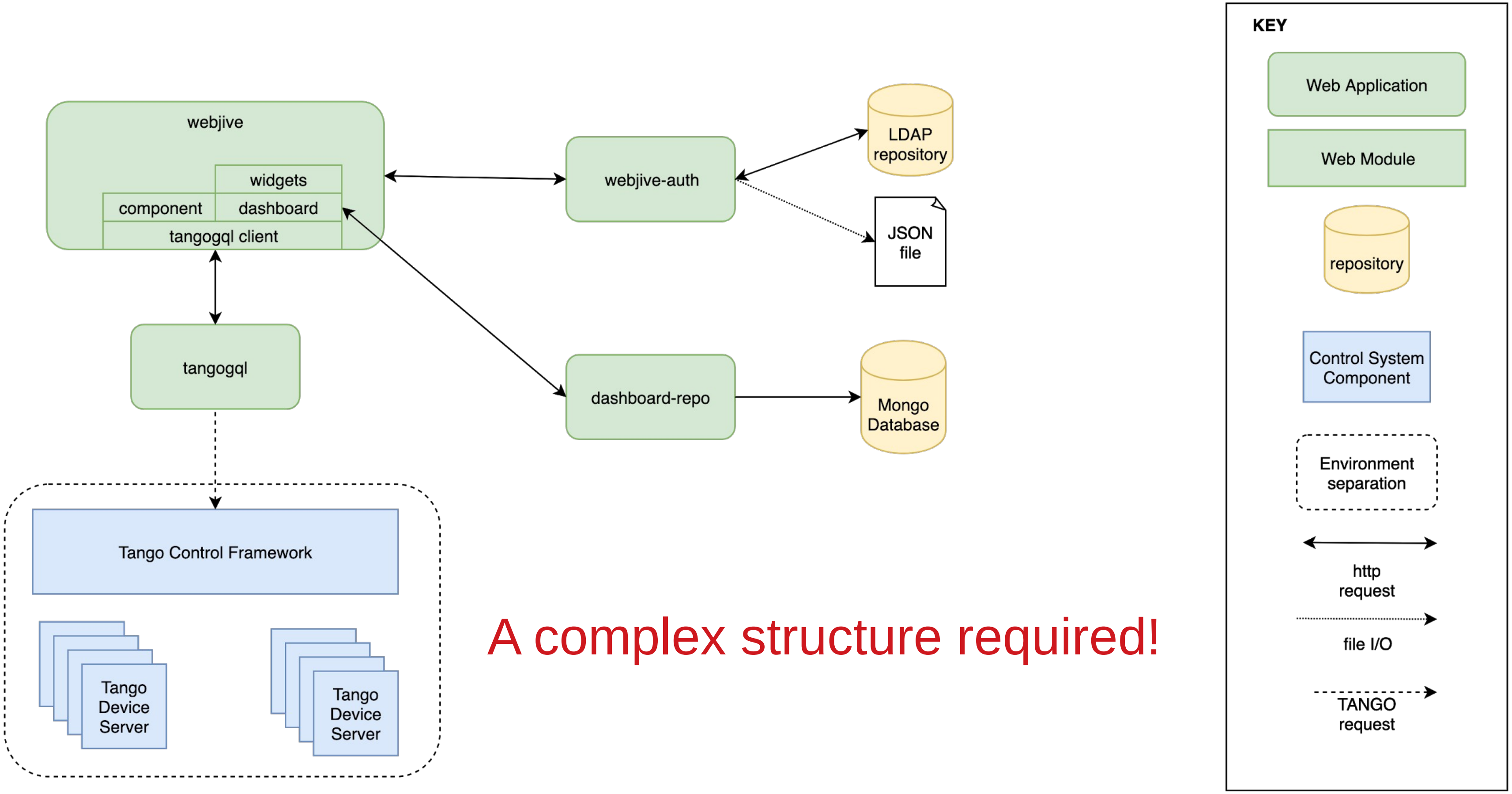


CSP Software Structure

A complex structure required!

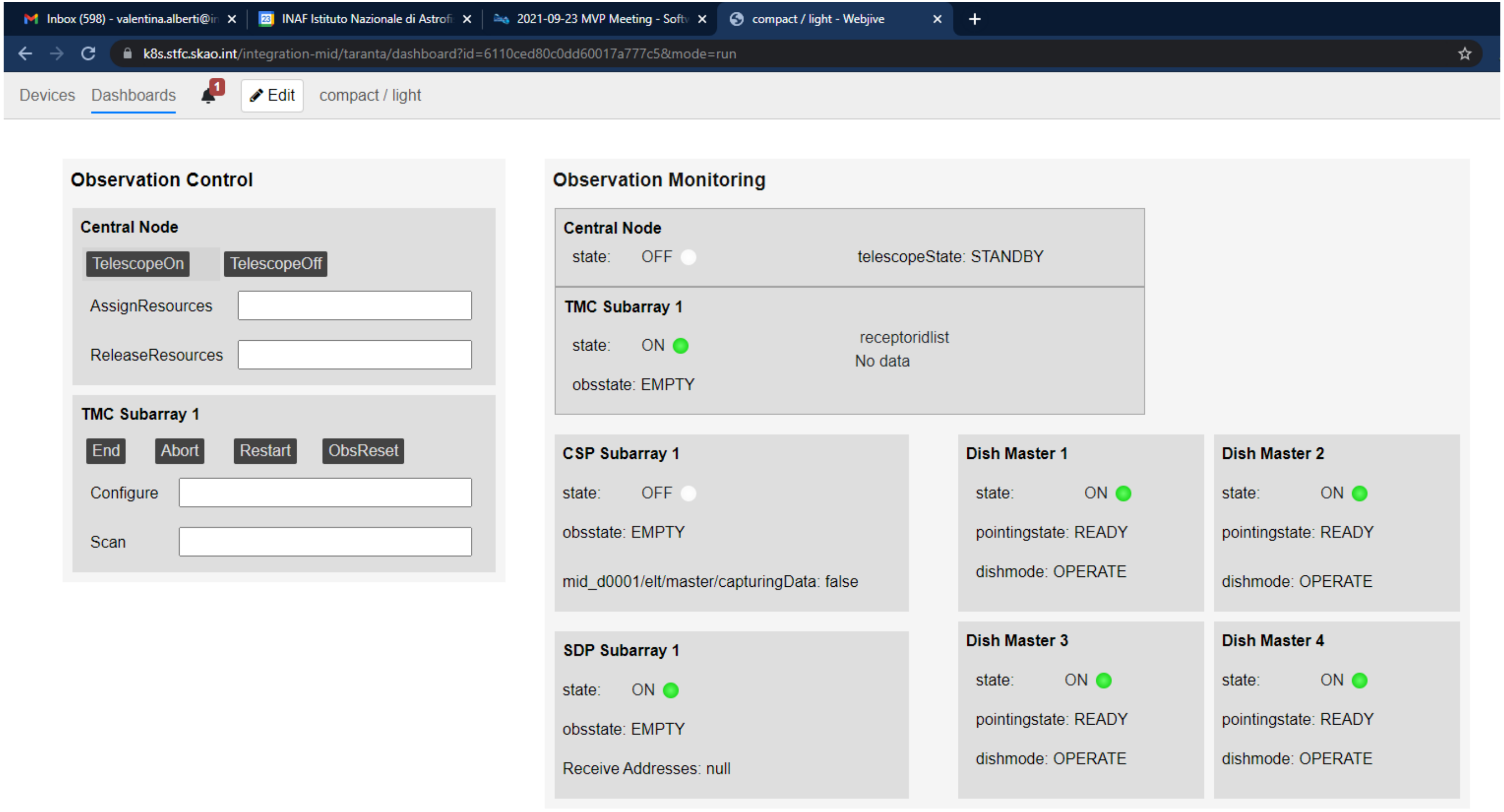


Taranta: a GUI infrastructure

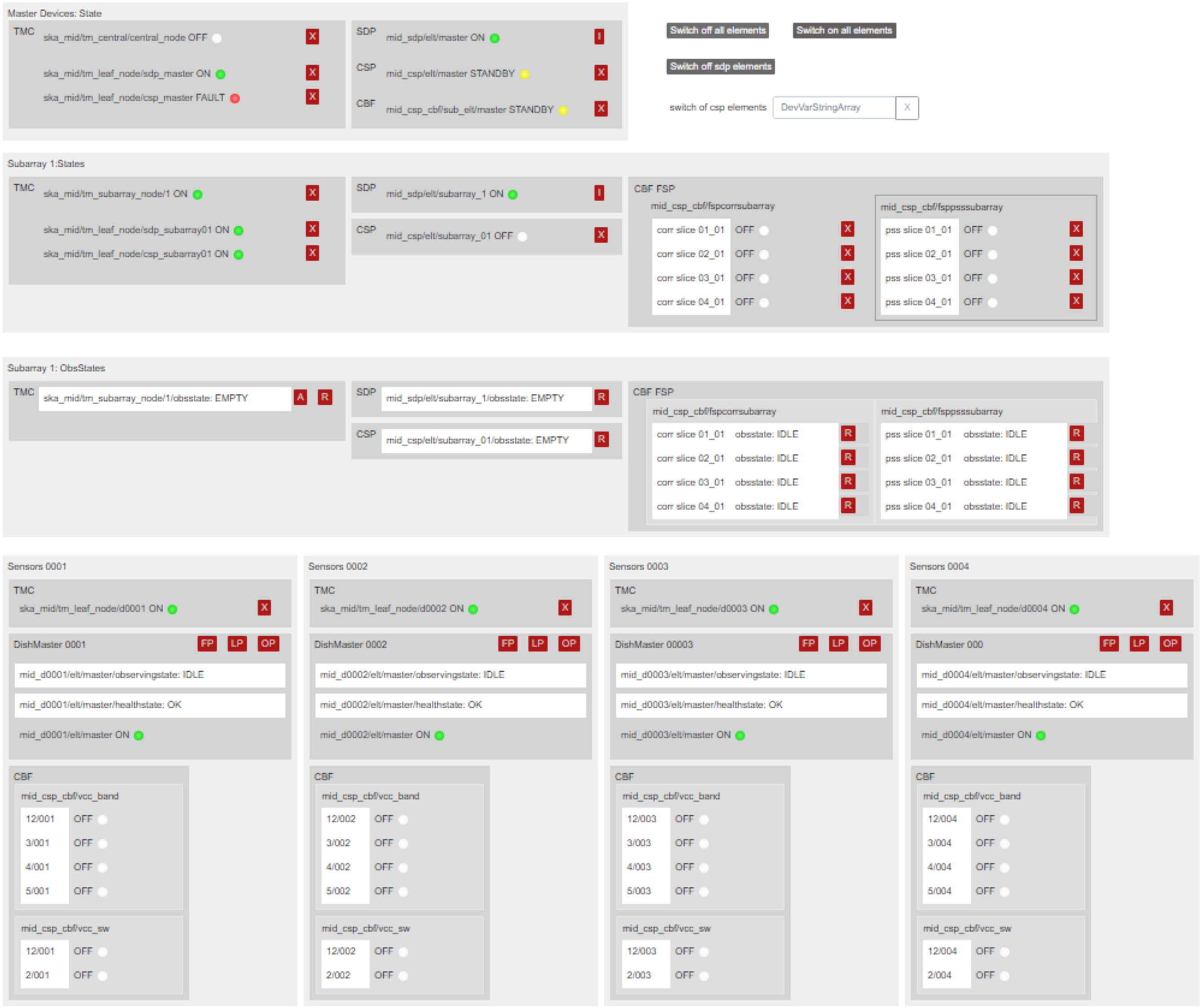


A complex structure required!

Taranta Example



A more Complex Taranta Example



Thank you

ISTITUTO NAZIONALE DI ASTROFISICA
NATIONAL INSTITUTE FOR ASTROPHYSICS