CTAO
Cherenkov Telescope Array Observatory

GOBIERNO DE ESPAÑA
MINISTERIO DE CIENCIA E INNOVACIÓN
Ciemat
Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas

Array Operations

CTAO School - 2nd edition

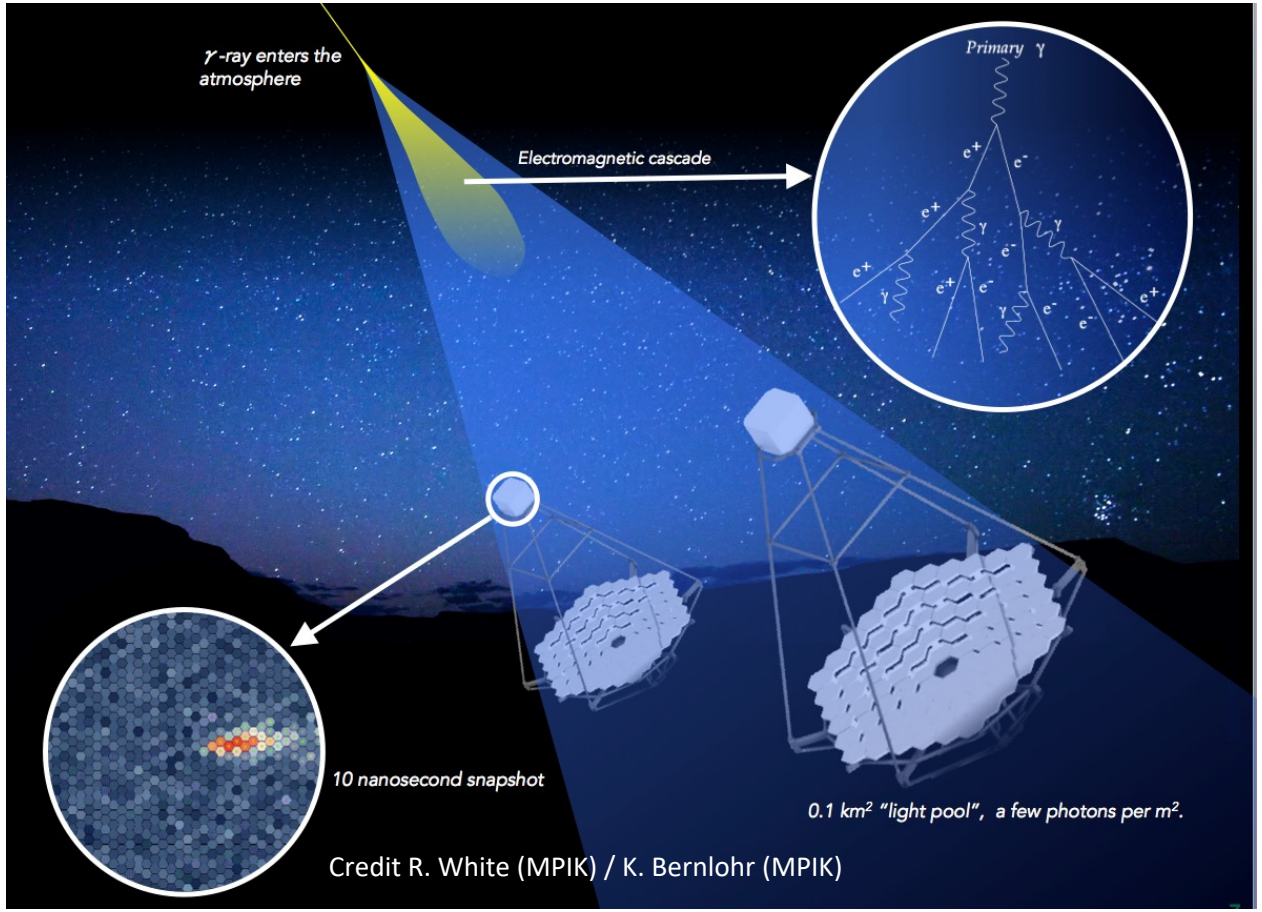
Igor Oya, CIEMAT.
CTAO ACADA Coordinator, Computing Deputy Coordinator

Summary

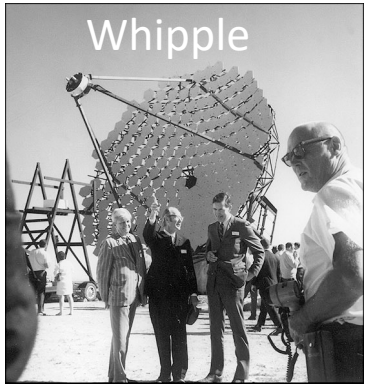
- Intro – what is an array of Cherenkov Telescopes?
- The CTAO arrays – day an night operations, and its crew
- The CTAO system composition as arrays of Cherenkov Telescopes
- Using the CTAO Arrays for acquiring science data
- Array Operations – the Software perspective

Cherenkov Telescopes

IACT: Imaging Atmospheric Cherenkov Telescopes (or Technique)

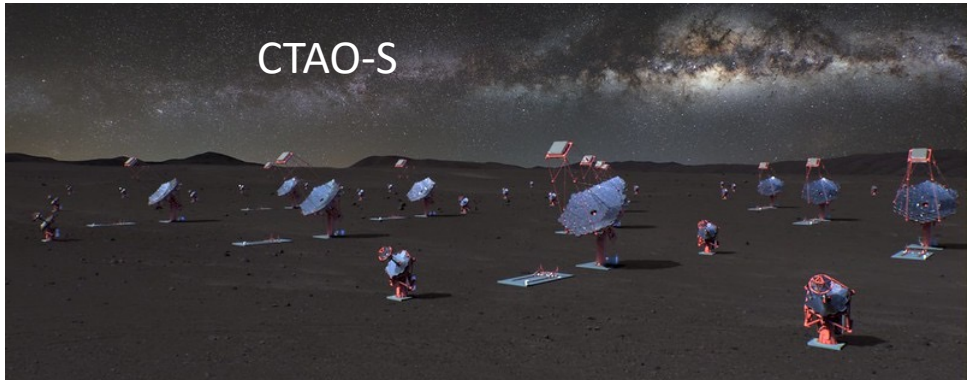


- Cherenkov telescopes can be operated individually – and successfully.
  - See the examples of the Whipple telescope, MAGIC 1 until 2028, LST1 last years.
- But the full potential of the IACT are exploited when operated as arrays.

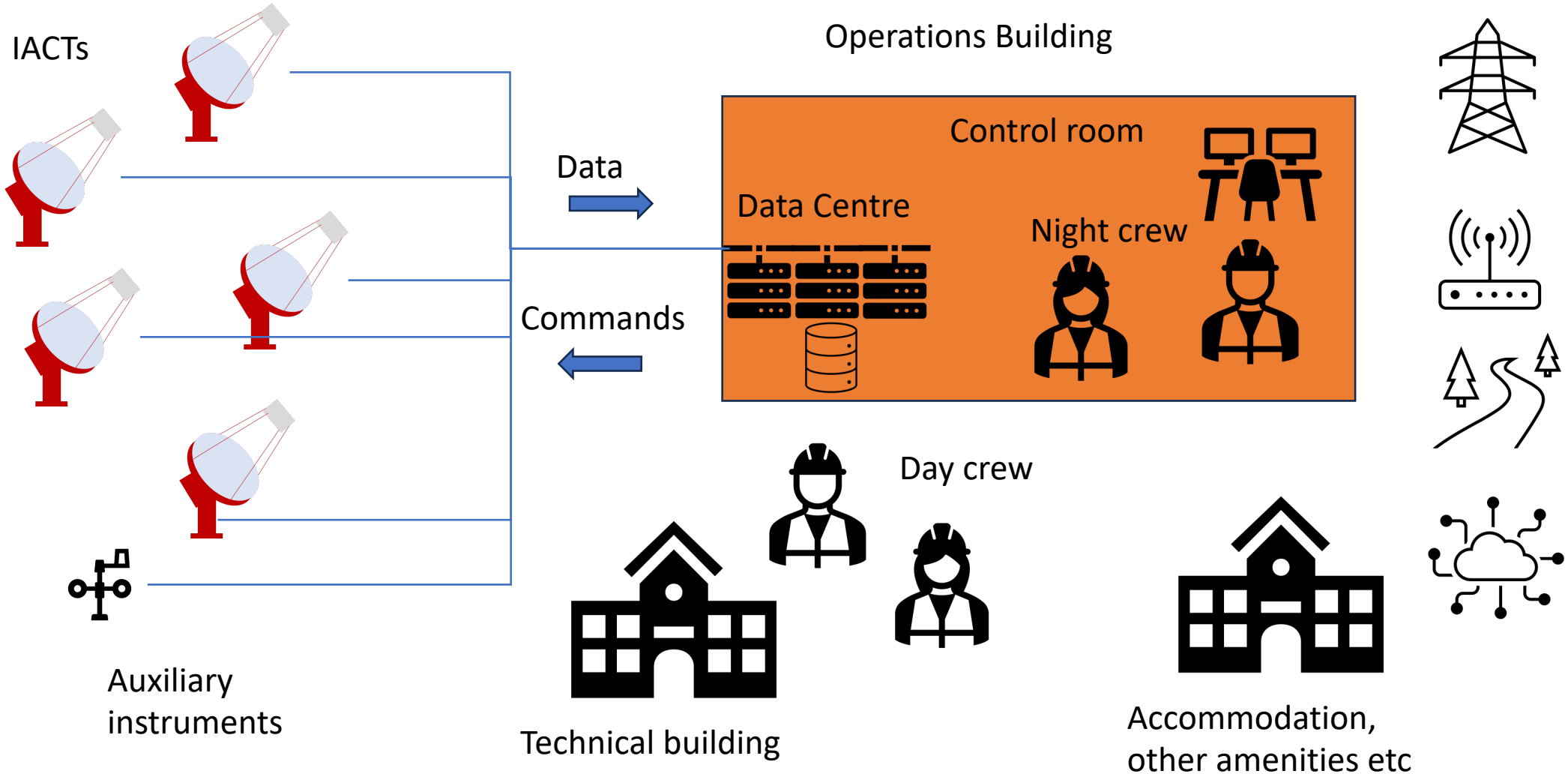


Arrays of IACTs

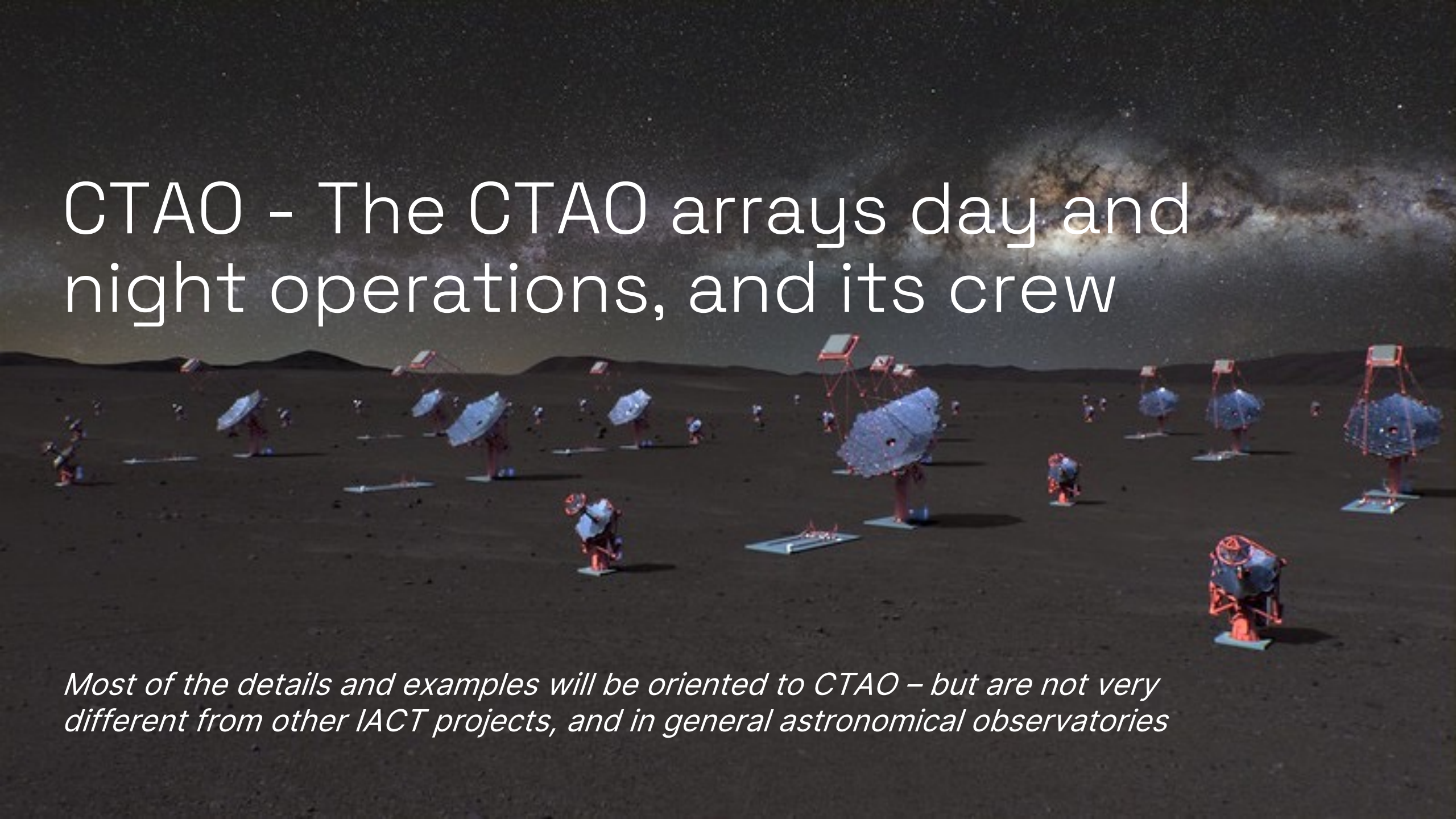
- All IACTs projects of the the very successful 3rd generation are all arrays of telescopes.
- All of them include auxiliary instrumentation, such as weather stations, LIDARs, etc. to monitor the atmosphere.
- CTAO is built on top of the success of the 3rg generation IACTs, and will have two arrays of telescopes, one in La Palma (Spain) and another one in Paranal (Chile).



Anatomy of an IACT array

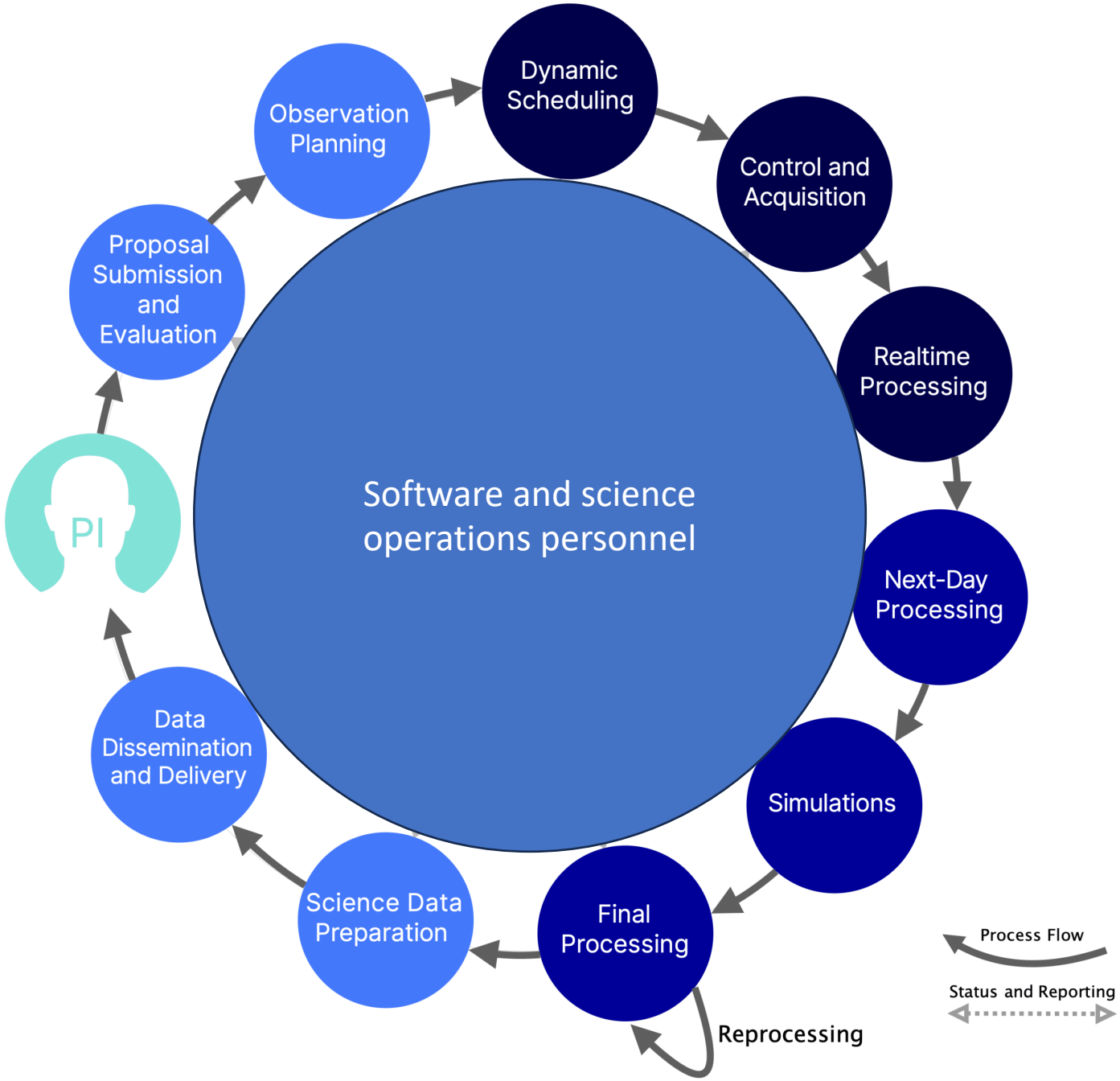


CTAO - The CTAO arrays day and night operations, and its crew



Most of the details and examples will be oriented to CTAO – but are not very different from other IACT projects, and in general astronomical observatories

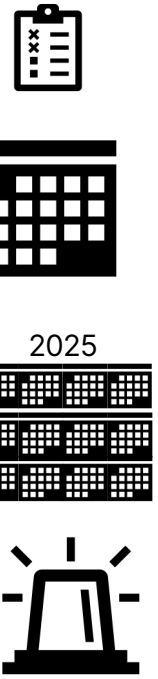
The science operations of CTAO



Array operations

Timescales

- Daily
- “Monhtly” (per moon period)
- “Yearly” (per observation cycle)
- Unpredictable (or at undertemined times)



Daily operations

- The main business of an observatory is to acquire scientific data, and for an IACT it means to observe during dark nights – to observe as much as possible!
- To optimize scientific return of investments, and other KPIs, we want our instruments to work for as much time as possible during dark time.
  - Optimize the the availability
  - This requires a lot of activities must happen during the days, so the instrumentation is up and running every night
- Some operations happen at 24/7/365, such as gathering of instrument health data, weather stations etc.
- An ICT array, and any Observatory, is busy with activities happening at the sites, day and night.

Availability = uptime/(uptime+downtime)
KPI = key performance indicators

Day and technical operations

(Passive) Day Operations:

- Gather instrument monitoring information: sensors, status.
  - important to detect upcoming problems (preventive maintenance), or understand why certain problems occurred, and fix them as soon as possible (corrective maintenance…).
  - Display status and system health for the day or night crew with Human Machine Interfaces (HMI)
- Reception of external science alerts (see later), to be included in the night observations.

Technical (day and night) operations

- In-situ instrument maintenance actions: preventive or corrective maintenance.
- Replacement of instrumentation, upgrades, etc.
- In-person or remote maintenance, calibration, and test actions, some of them to be done during the night and others during the day.
- Data Centre maintenance.
- Software upgrades, regression testing, …
- Supervision of data transfer from the sites to and from offsite data centres.

- Other important activities are omitted here: maintaining the buildings, cleaning, water, access control, etc.

Day Crew

- The day crew is composed of engineers, technicians, instrument scientists and other professionals that make sure the telescopes and other instruments are in perfect shape for the next night.
- I do not mention them too much in the rest of the presentation, but their work is essential for keeping the observatory running.

Credit: Daniel López / IAC

Credit: Daniel López / IAC

Science (night) operations

Science (night) operations:

- System preparation for observations, regular calibrations, checks at startup.
- Prepare scheduled observations of the night (automatic or manual).
- Special calibration operations (e.g., external light sources).
- React to external or internal science alerts.
- System health, data quality monitoring, system monitoring.
- Address other unplanned requests or situations (see later).

Notes:

- Observations are split into atomic units of ~20-30 minutes (“runs”, or “Observation Blocks”).

Night Crew

- In many IACT (and other type of instruments), the executions of the observations are performed by a crew of rotating shifters, often being members of the scientific collaborations that built the instrument.
  - Typically, shifters spend one month cycle in the sites, however some experiments have implemented remote observation procedures.
  - Joining an IACT as shifter is a great opportunity for young researchers to learn how the systems work but also creates some burden in the collaborations.
- In some projects, and in particular in more traditional astronomical fields, “professional”, hired operators, sometimes also astronomers on duty (or support astronomers), are responsible for performing the observation.
  - CTAO will operate with professional night crews.
- Shift work is critical work for the observatory operations, and having a dedicated night crew can significantly improve the KPIs of the Observatory.

Night Crew: Operator and Support Astronomer

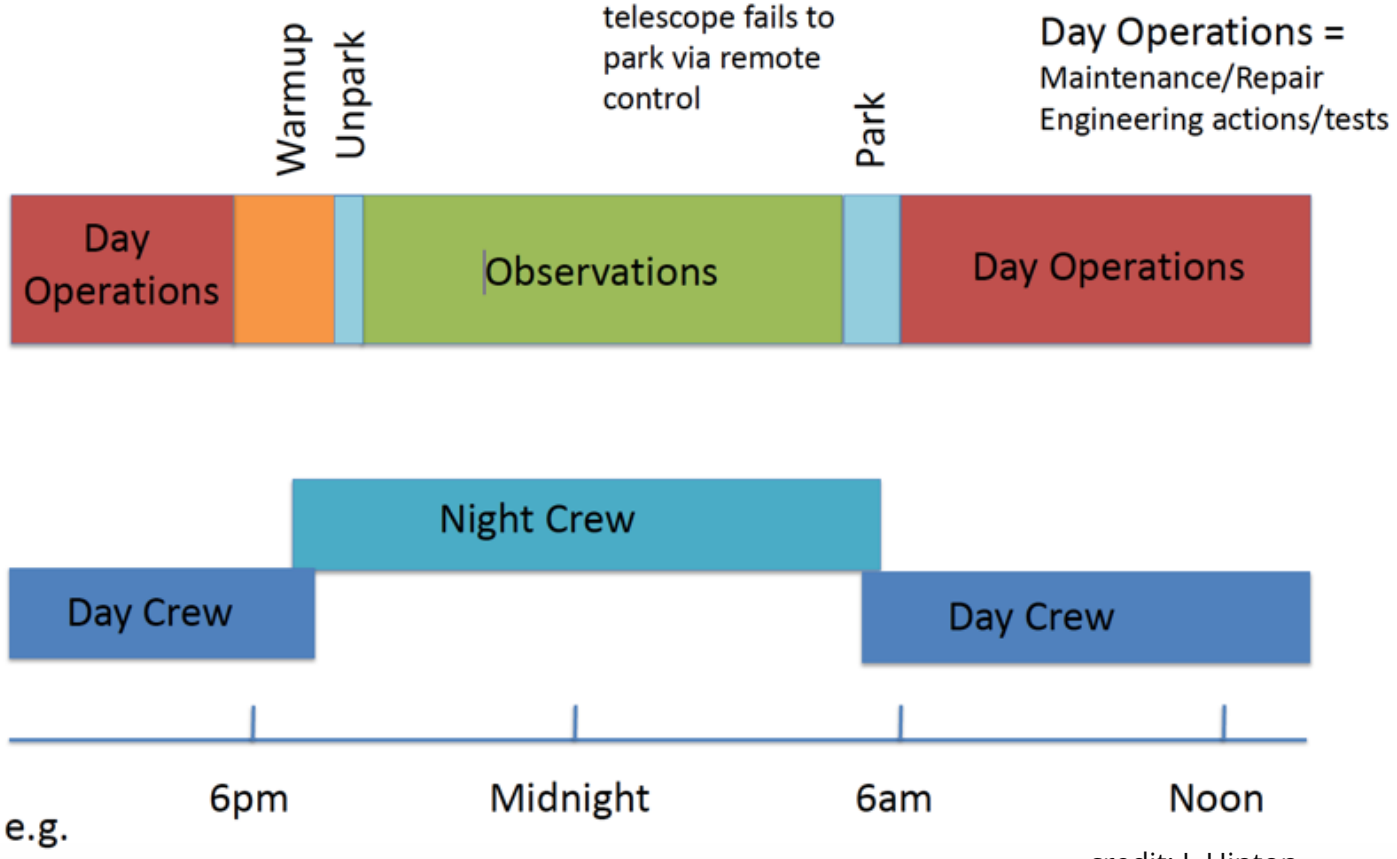
- In CTAO, we have identified the need for two individuals during the night in the control room, with the following roles:
  - **Operator:** Engineering and technical aspects. Responsible for supervising and carrying out scheduled observations and calibrations during the night. Troubleshoots problems, can modify the schedule, if necessary (e.g., weather, Science Alerts), and logs all activities.
  - **Support astronomer:** Scientific aspects. Oversees and supports scheduling of observations, supervises reactions to external and internal science alerts, and checks science analysis and data quality monitoring results.
- This distribution of roles assumes dedicated Human Machine Interfaces (HMI) and aim to optimize the operation costs of the observatory.
- Other Observatory personnel, either at the sites or elsewhere, available to act if the night crew requires urgent support. For non-urgent issues, the day crew takes over the day after.

Day & night operations

Warmup =
Telescopes parked, sub-systems to Standby

Park =
Normally rapid transition of all elements back to Safe state, opportunity for manual intervention if telescope fails to park via remote control

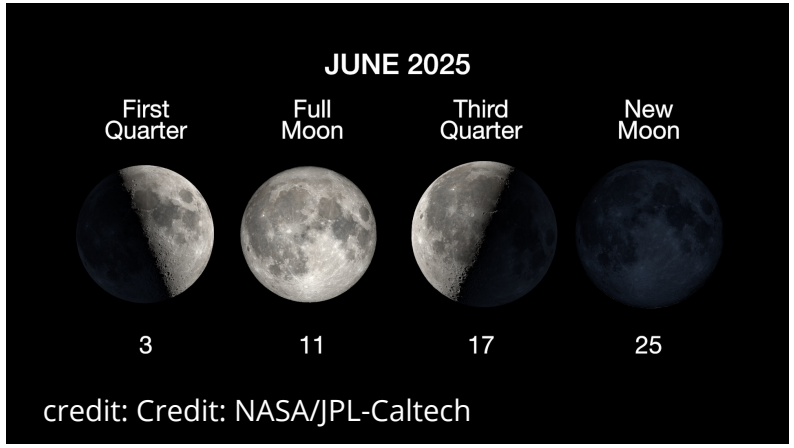
Day Operations =
Maintenance/Repair
Engineering actions/tests



credit: J. Hinton

Moon cycle and IACTs

- Typically, IACTs do not take data the days closer to full-moon days, and during the time of the night when the Moon is too bright.
- Some IACT projects have found schemes to observe even while the moon is relatively bright.
- The full moon days are the natural time to switch night shift crew, or to perform maintenance activities, such as software upgrades.
- Having a few days without taking much data also helps us reduce the computing storage costs and data link.



Yearly cycle

- Many astronomical projects prepare the observations in yearly cycles.
- The observation of the “next cycle” is planned and carried over during the cycle.
- Many observation programs require several hours, some of them can even require more than a hundred hours.
- Some CTAO’s key science programs (KSP) will require hundreds of hours observations and will be spread over several cycles
- The latitude of the observatory influences the length of the observing night depending on the season.
- Certain calibration and preventive maintenance activities need to be performed on a yearly basis
- One could expect up to 1400 hours/year of dark observation time, but with two sites and subarry operations, CTAO will be able to accomodate more proposals every cycle.

“Unpreditable”: Good and bad surprises

- **Science Alerts** (Transient events, ToOs). An unplanned event of scientific interest that potentially requires fast updating the current schedule

- **Alarms:** A problematic situation of the system that requires action of the Night or Day Crew for mitigation

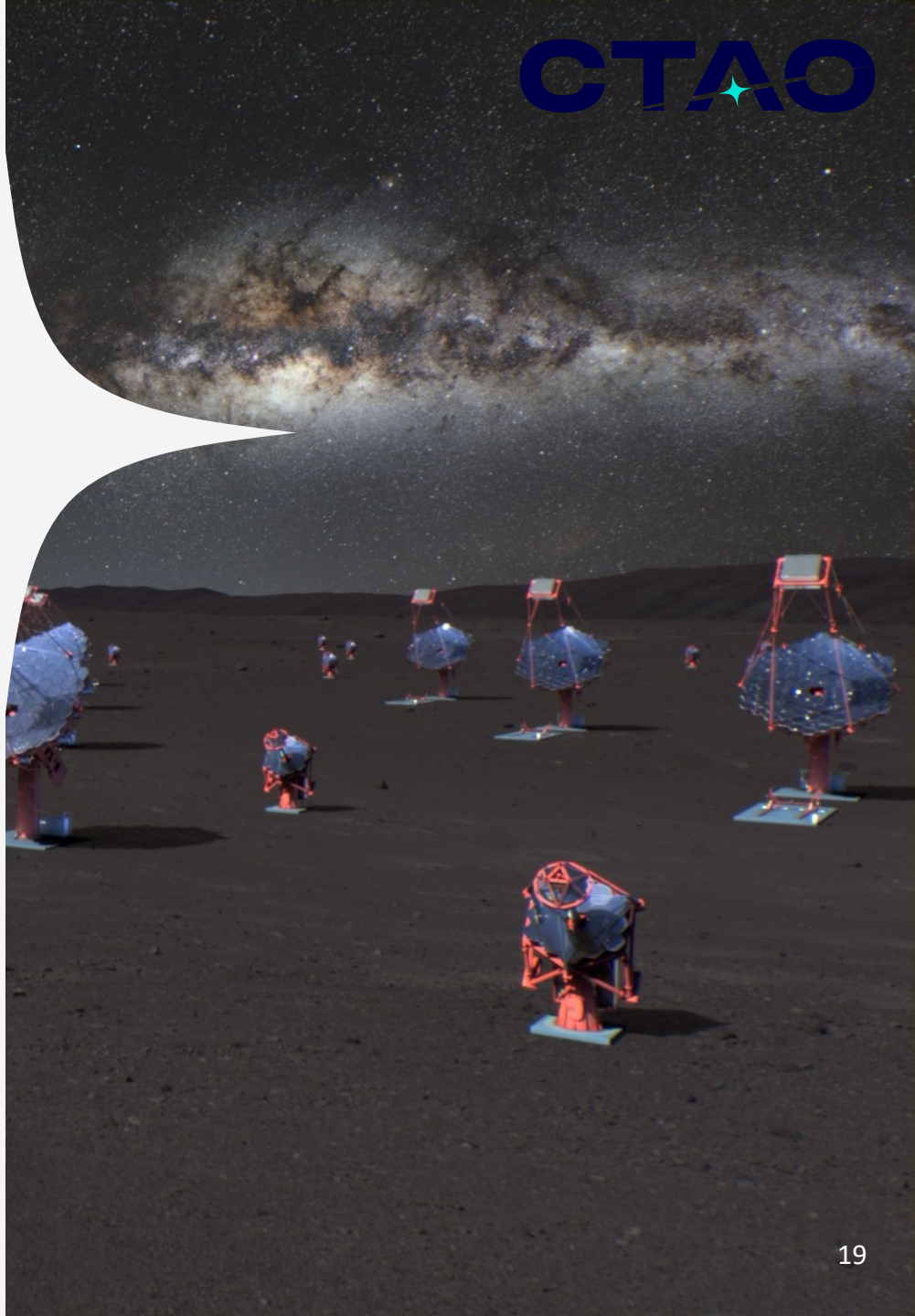
In both cases, the expertise of the night crew and the adequacy of the software systems are critical to optimize the scientific return of the Observatory.

Credit: NASA/Swift/Cruz deWilde

Credit: BrokenSphere/Wikipedia

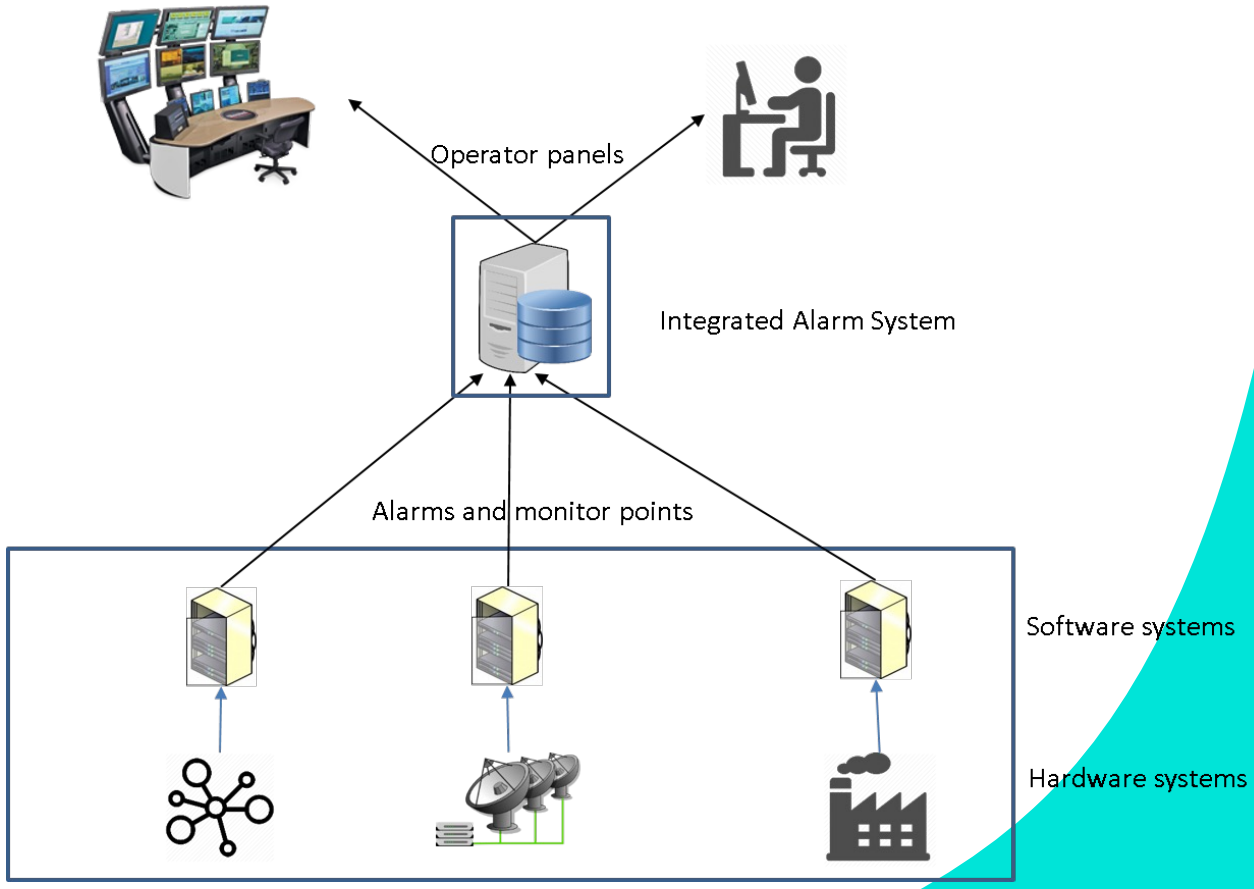
The CTAO Transients Handling

- With a relatively small field of view (< 10°), CTAO will rely on external alerts (e.g. from IceCUBE, LIGO, Fermi-LAT) to follow up on targets of opportunity (ToO)
- CTAO is expected to receive a few thousand of these Science Alerts per night
- Self-triggered alerts from the real-time analysis are also supported
- The Transients Handler is expected to process, schedule, and start observations after a few seconds of receiving an alert



Alarms

- Alarms are exceptional situations that require operator action.
- A telescope may stop responding. Or a device may become overheated. Or disk storage may be getting full. Or power is lost from a group of telescopes.
- Alarms have severity levels: From very serious, safety related alarms, to relatively minor alarms that can be “shelved”.
- Efficient addressing of alarms is critical for the Observatory KPIs.



Ref.: ESO ALMA Support CentreIntegrated Alarm System Design Document Number: ESO-299387

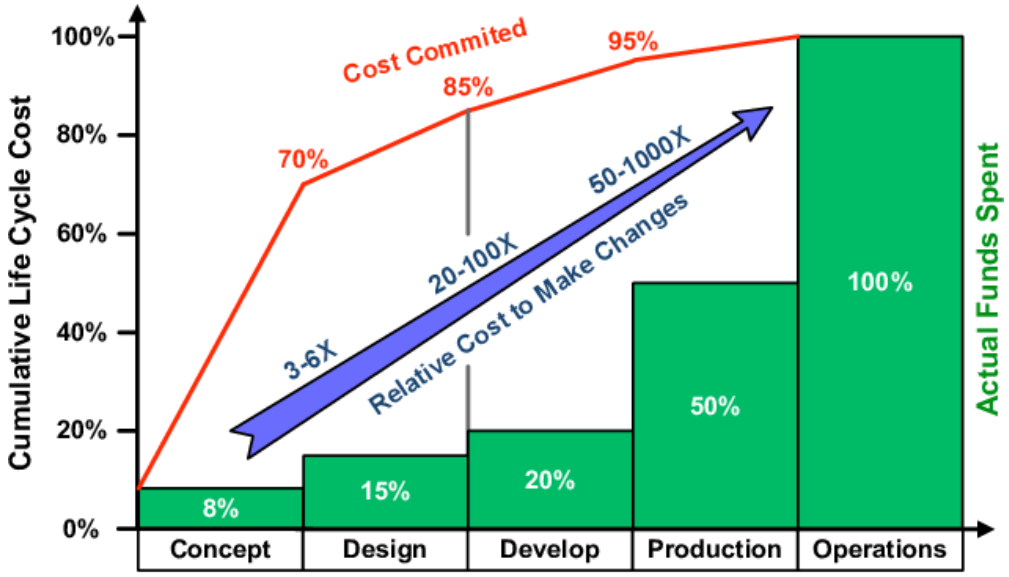
Being a night crew member- excersise

- Can you picture being a night crew member? What are the most important things for you to do a good job?

The CTAO system

Systems engineering

- “Systems engineering” is defined as a methodical, multi-disciplinary approach for the design, realization, technical management, operations, and retirement of a system.
- A “system” is the combination of elements that function together to produce the capability required to meet a need.
- The elements include all hardware, software, equipment, facilities, personnel, processes, and procedures needed for this purpose; that is, all things required to produce system-level results.
- Any large-scale project, incl. scientific installations, benefit from following the technique.

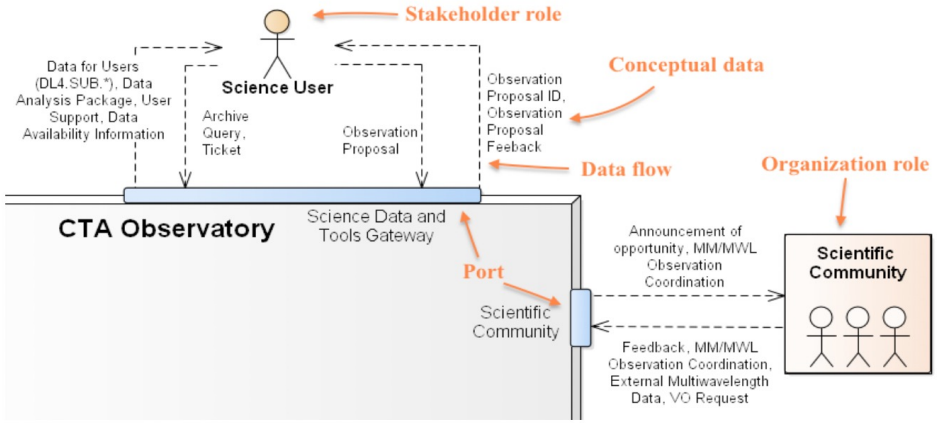


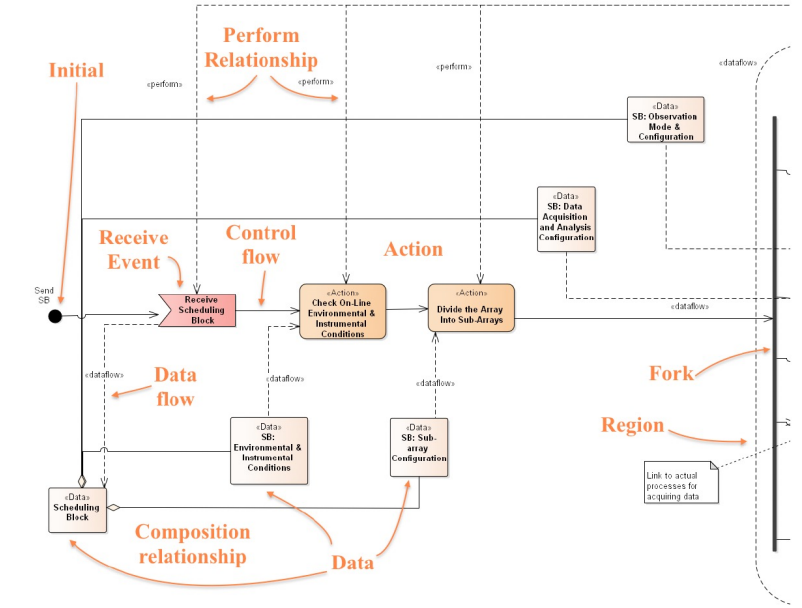
© A. Sanders, J. Klein

https://www.nasa.gov/reference/2-0-fundamentals-of-systems-engineering/

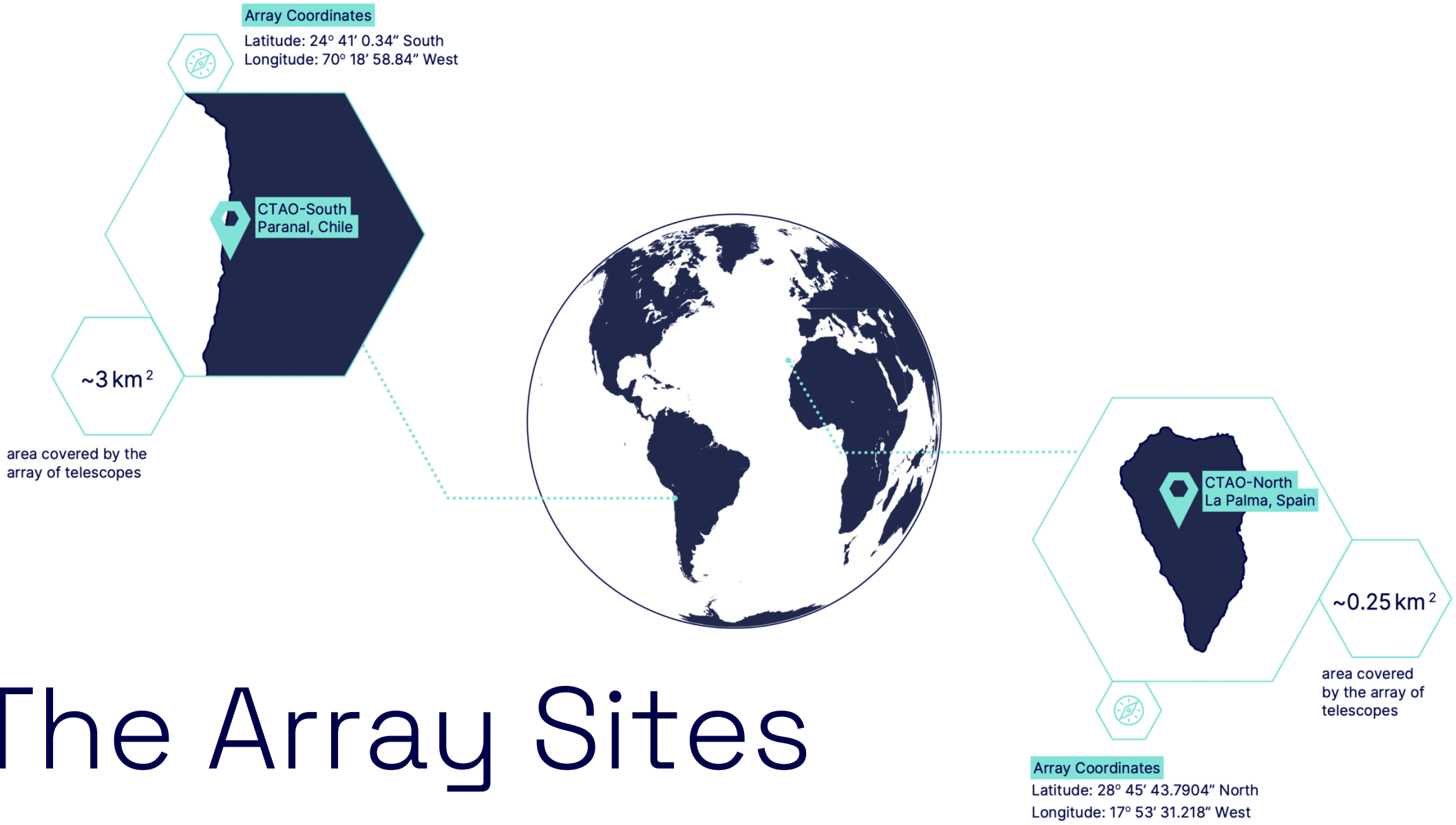
SysML – architecture modelling

- The systems modeling language (SysML) is a general-purpose modeling language for systems engineering applications. It supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems.
- Connected with the Unified Modelling Language (UML), often used in the software world.
- For large and complex projects such as CTAO, modeling techniques offer great advantages.
- Used in large projects, incl. astronomical projects.
- We uses SysML for CTAO's design.





The Array Sites

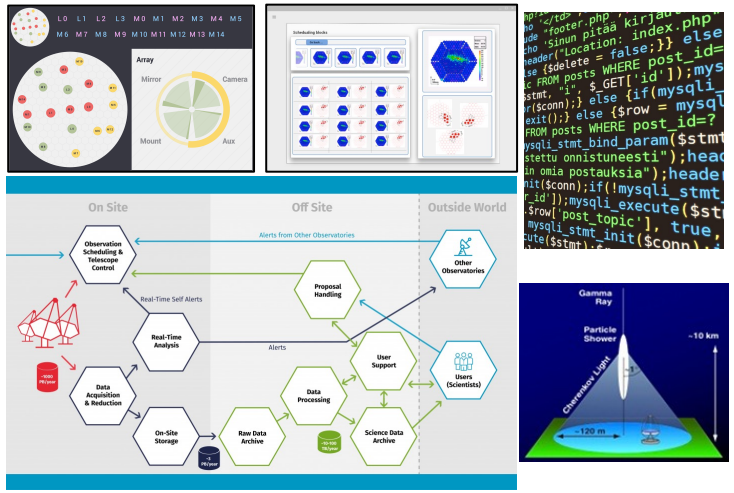


CTAO Systems

The CTAO architecture was defined following Model-based systems engineering approach

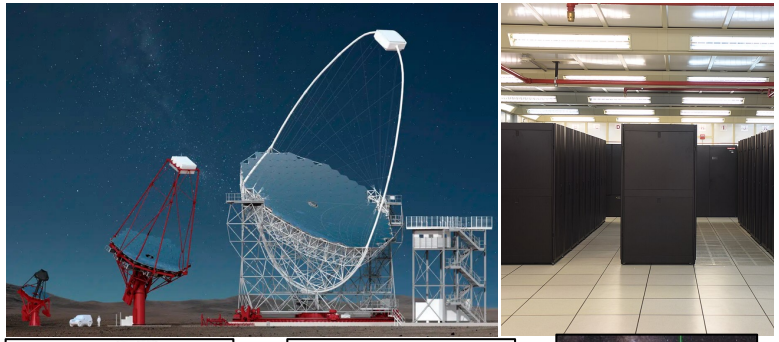
Science Operations

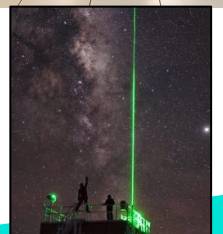
- Array Control and Data Acquisition (ACADA)
- Data Processing and Preservation System (DPPS)
- Science User Support System (SUSS)
- Science Operations Support System (SOSS)
- Authorization and Authentication (AAI)



Technical Infrastructure

- Telescopes
- Integrated Protection System (IPS)
- Array Calibration and Env. Mon. System
- Array Infrastructure Elements (power management etc.)
- On-site ICT
- Off-site ICT
- Clock System (CLK)





Observatory Operations and Administration

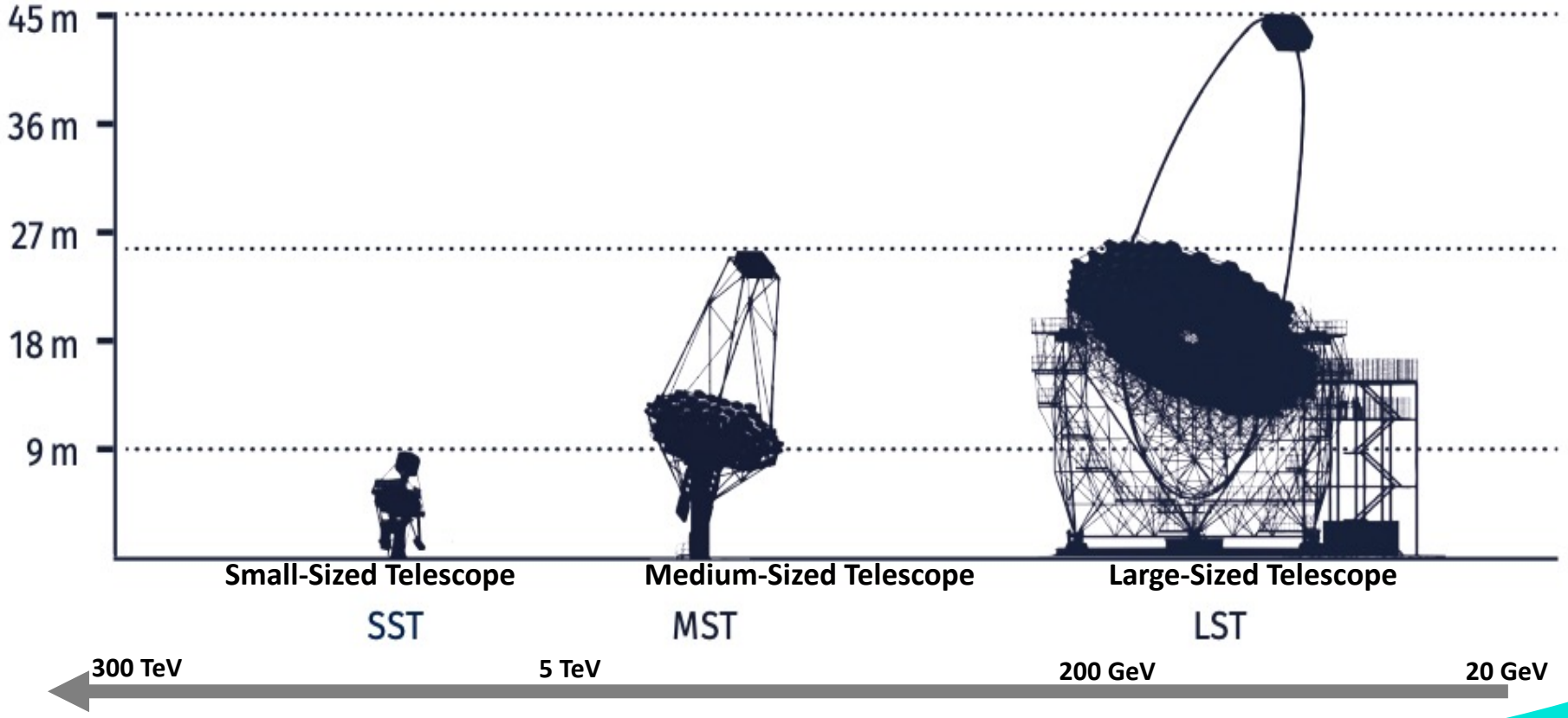
- Technical Operations Support System (TOSS)
- Management and Administrative System
- Office IT
- …

3 telescope designs

- 2-mirror Schwarzschild-Couder optical design
- 4.3 m ⌀ primary reflective surface
- SiPM camera: 2048 pixels (0.16°)
- 8.8° FoV
- 17.5 tonne

- Davies-Cotton optical design
- 12 m ⌀ reflective surface
- PMT camera – 2 designs:
  - NectarCAM: 1855 pixels
  - FlashCam: 1764 pixels
- ~7° FoV
- 82 tonne

- Parabolic optical design
- 23 m ⌀ reflective surface
- PMT camera: 1855 pixels (0.1°)
- 4.3° FoV
- 100 tonne

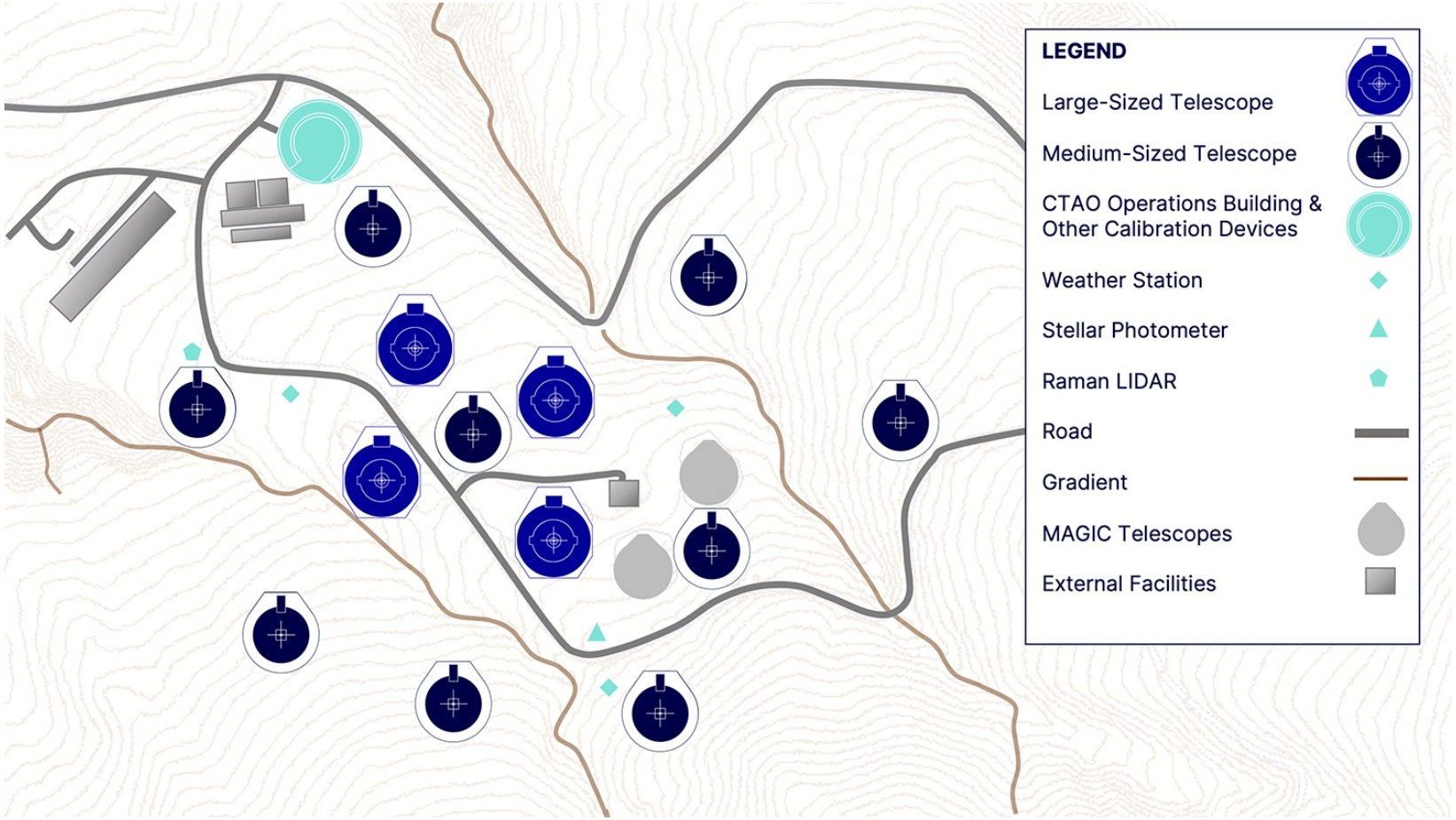


CTAO-North

- Canary Islands at an altitude of 2,200 metres above the island of La Palma
- Organized by the Instituto de Astrofísica de Canarias (IAC) at its Roque de los Muchachos Observatory
- The LST-1 is in service and three more are under construction (expected to be operational by 2026)

Watch the film about CTAO-North.

CTAO-N – Planned array

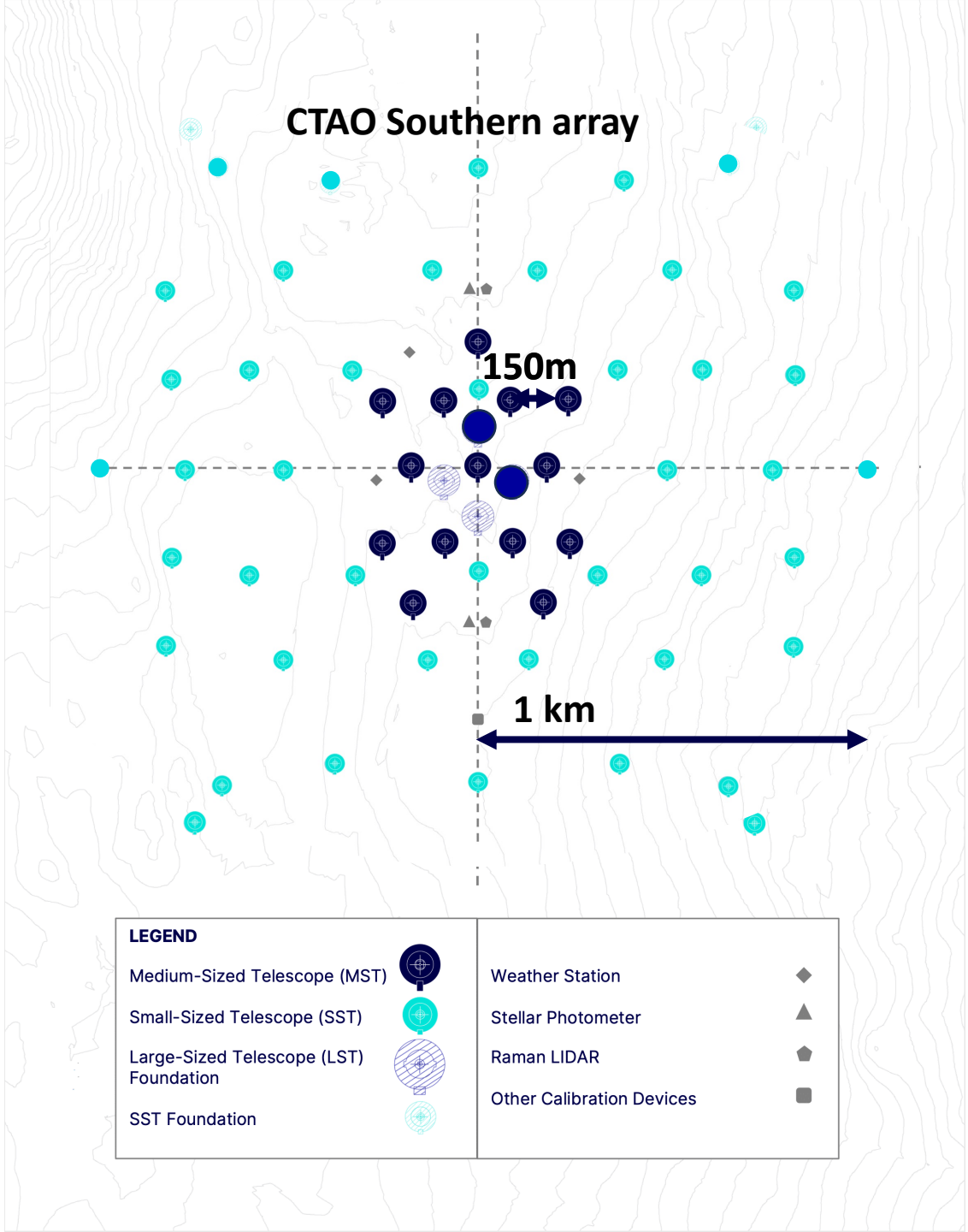


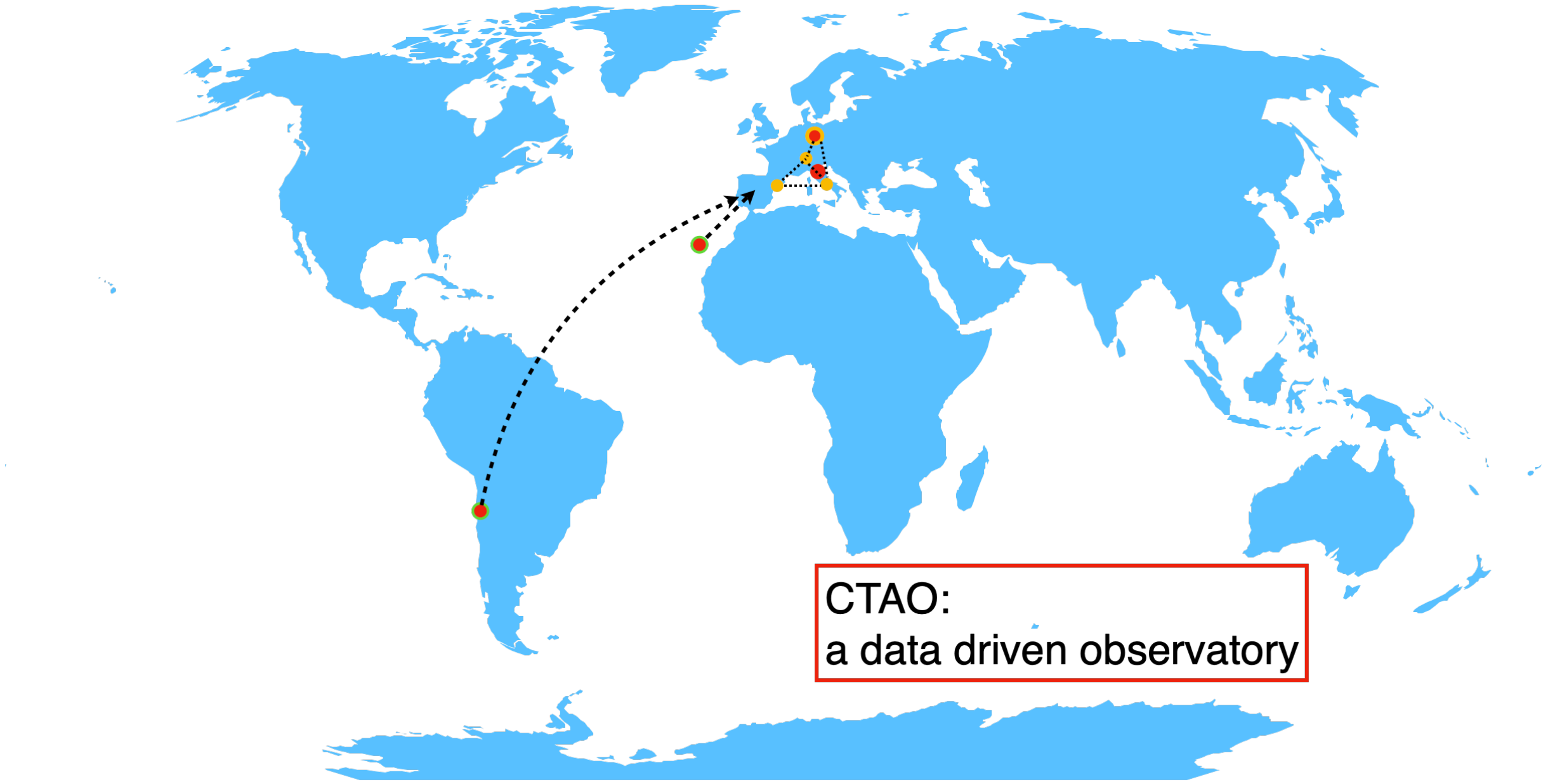
CTAO-South

- Located in the Atacama Desert in northern Chile
- Organised by ESO near its Paranal Observatory
- At present, only infrastructure works, including roads and electrical systems, are being carried out
- But we expect it be full with telescopes soon!

Watch the film about CTAO-South.

CTAO-S


CTAO Southern array
150m
1 km
LEGEND
Medium-Sized Telescope (MST)
Small-Sized Telescope (SST)
Large-Sized Telescope (LST) Foundation
SST Foundation
Weather Station
Stellar Photometer
Raman LIDAR
Other Calibration Devices


CTAO:
a data driven observatory

Using the CTAO Arrays

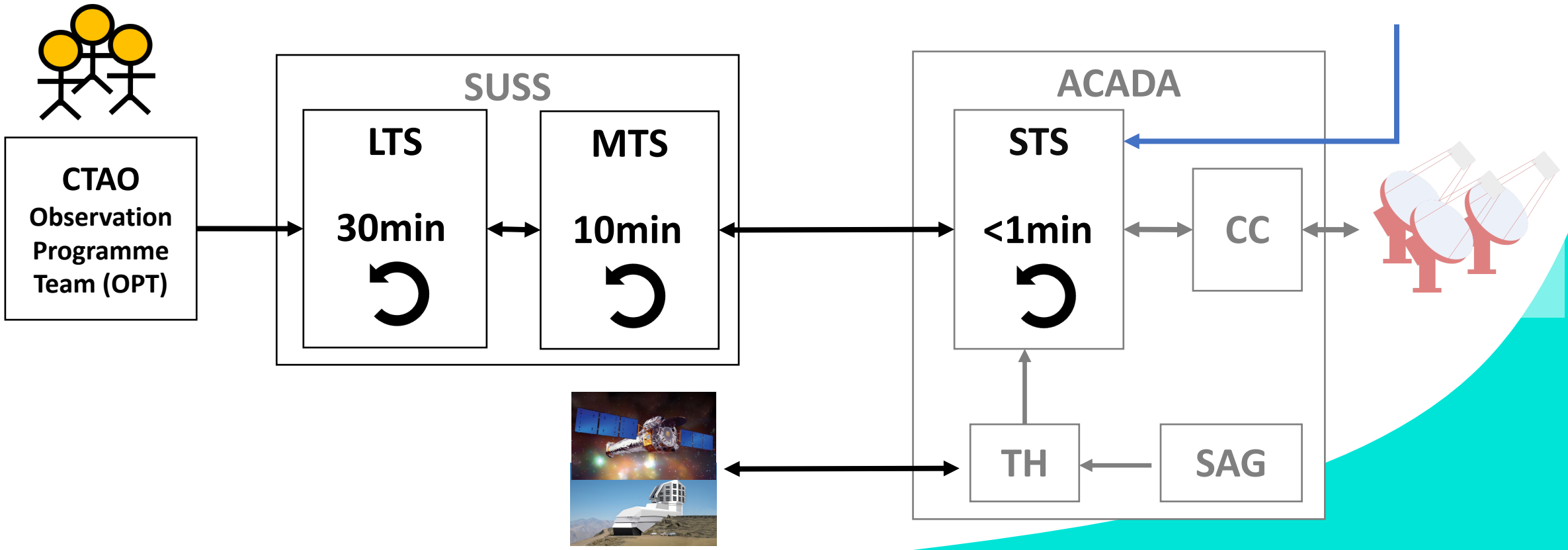
Using the CTAO Arrays

- Scientists will request observations with CTAO from a list of available configurations.
- Each configuration will offer different benefits.
  - A configuration with SSTs may be better suited to study the high-end energy spectrum of Galactic sources, and one with LSTs, the lower end of the spectrum of AGNs.
  - More telescopes together offer better sensitivity but may be overkill for certain goals.
  - There could be be more competition to obtain observation time with all telescopes of a site than a small fraction of them.
  - Some observations could be performed without perfect atmospheric conditions, and others not.
- Observations can be requested as target of opportunity observations.
- Different observation modes will be offered.
- Before each year cycle, the CTAO will announce the opportunities for observations and the offered configurations. Based on the received proposals and the outcome of the Time Allocation Committee (TAC), the schedule of the cycle will be optimized.

Scheduling

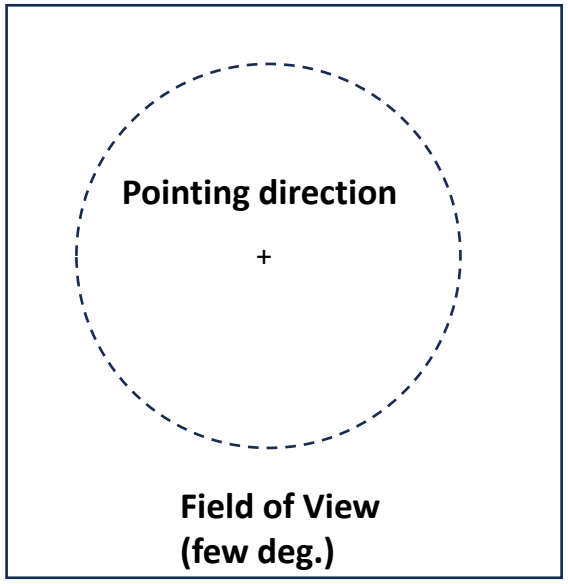
LST: Long-term scheduler
MST: Mid-term scheduler
STS: Short-term, scheduler
CC: Central Control
TH: Transients handler
SAG: Science Alert Generation pipeline (Real-time analysis)

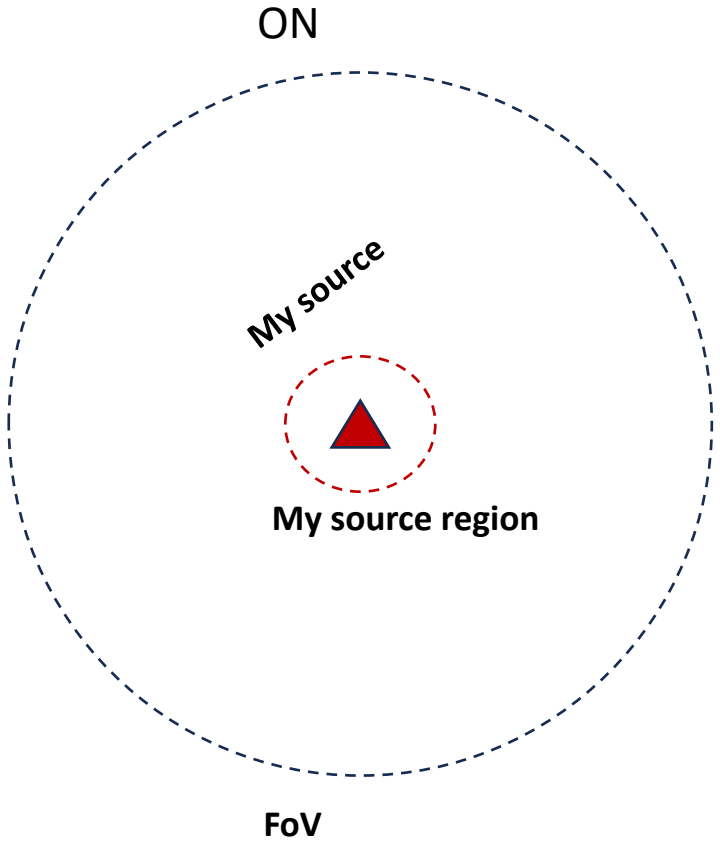
YEAR MONTH (up-to) NIGHT

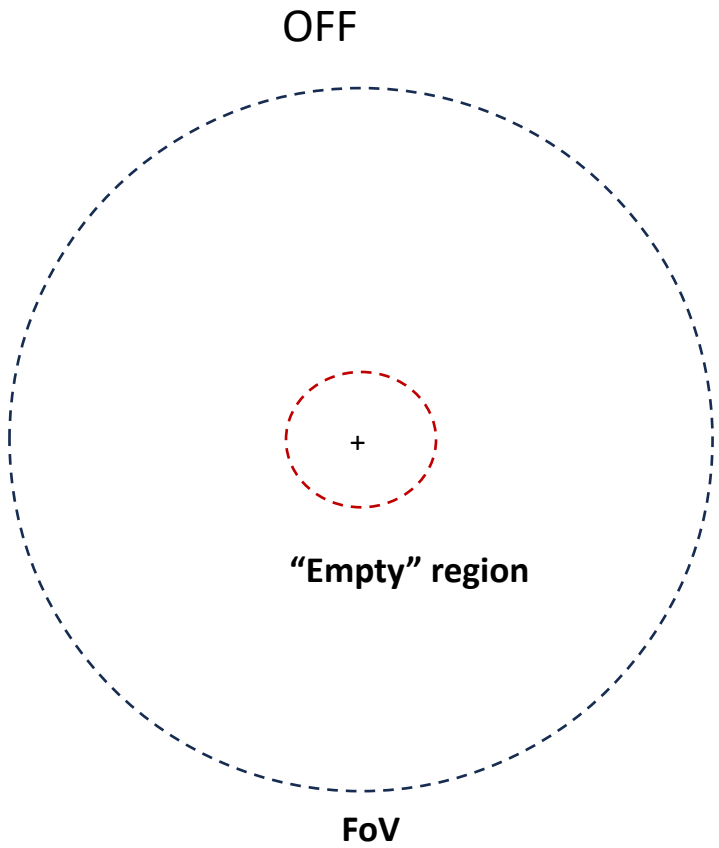


Observation modes: On-Off

- Take a series of “ON” runs and and of OFF runs. ON and OFF runs need to share similar characteristics.
- Signal calculated by subtraction of ON with OFF (normalizing the exposure).



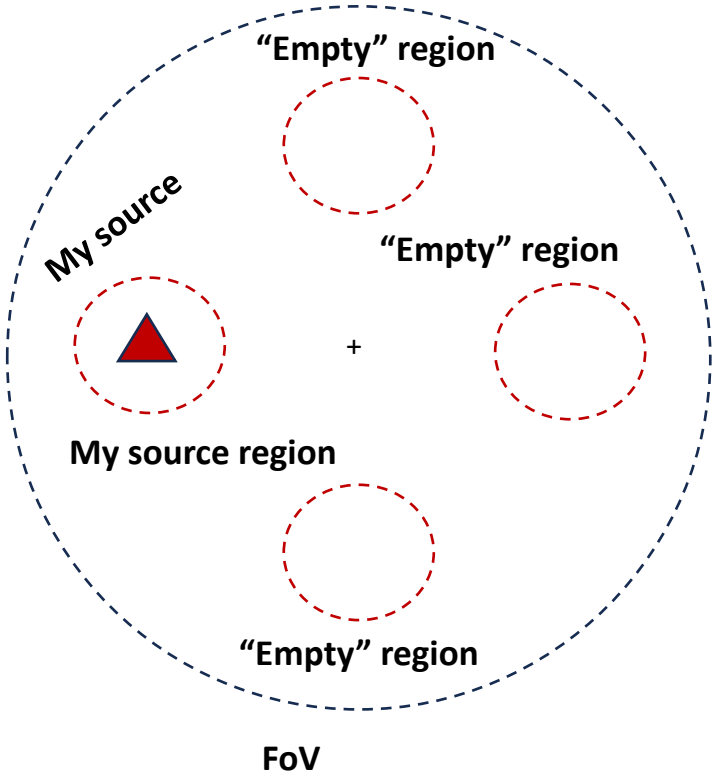




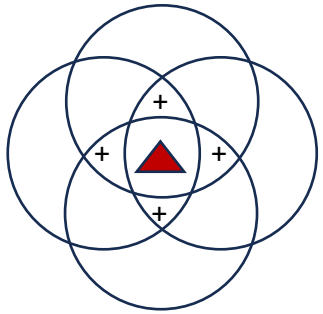
Observation modes: Wobble

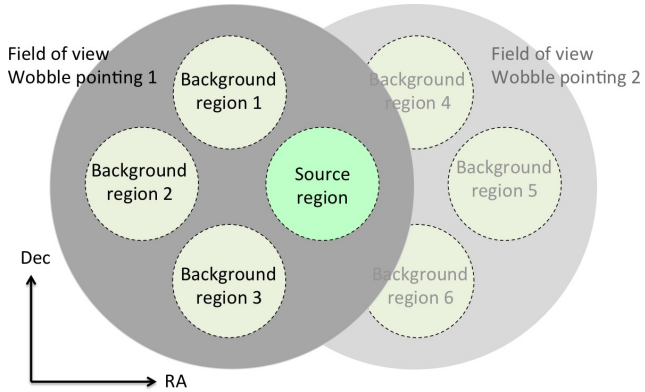
- Take a series of runs “wobbling” around the source.
- Signal is calculated by subtraction of ON (source region) with OFF (“empty” regions), normalizing the exposure.
- Alternate wobble positions to mitigate possible systematic errors (e.g., from camera inhomogeneities).
- No need to take dedicated OFF observations.

One “wobble”



Four “wobbles”

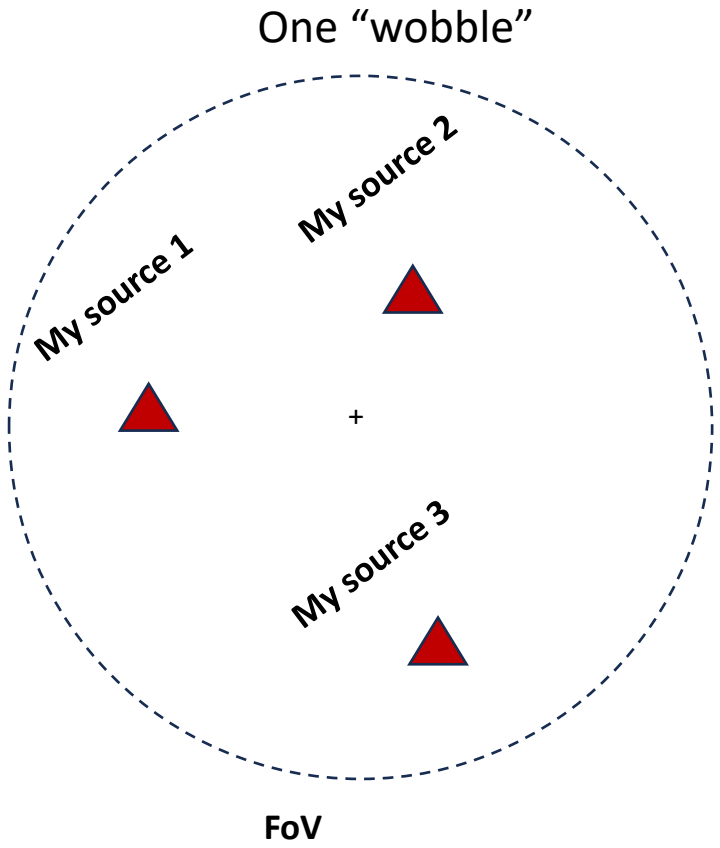




Credit: W. Hofmann

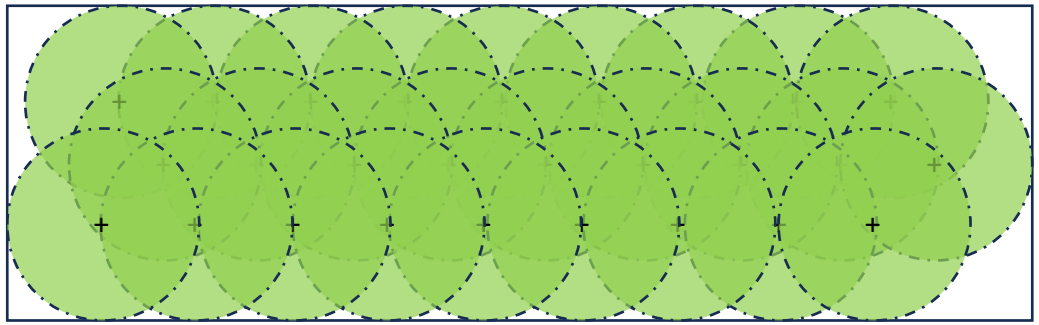
Observation modes: Wobble

- Note that with the large FoV of IACTs, often we would have more than one source candidates in the FoV.
- Some sources could be extended.
- Some bight stars may be in an off region.
- Wobble positions need to be selected with care.



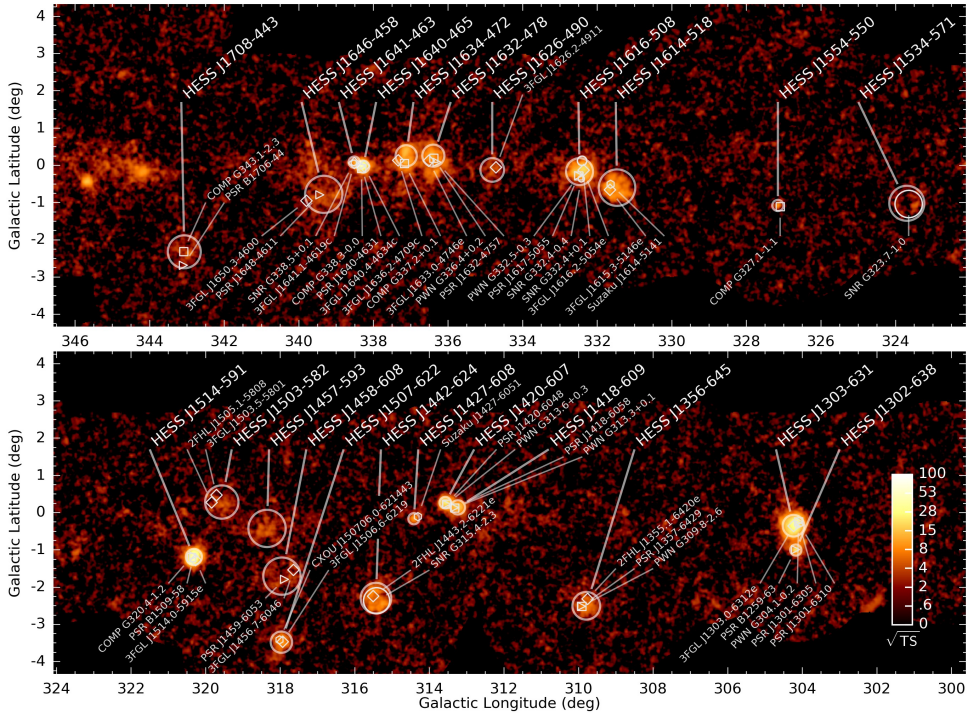
Observation modes: Grid/Survey

- Analogous to wobble, but we map a region with a series of overlapping observations
- Survey the Galactic plan, and other regions of interest



My region of interest

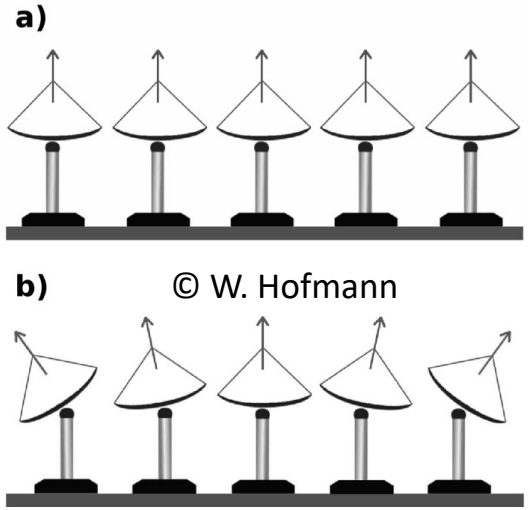
H.E.S.S. GPS



Observation modes: Variations

- Dift-scan mode: Points the telescope to an advanced in time position of the selected source and waits until it will pass through the field of view (FoV). By repeating this procedure many times, a significant amount of data can be collected

- Pointing modes:
  a) Parallel: all telescopes point towards thr same direction (standard)
  b) Divergent (experimental): slight offset of individual telescopes, for larger FoVs.

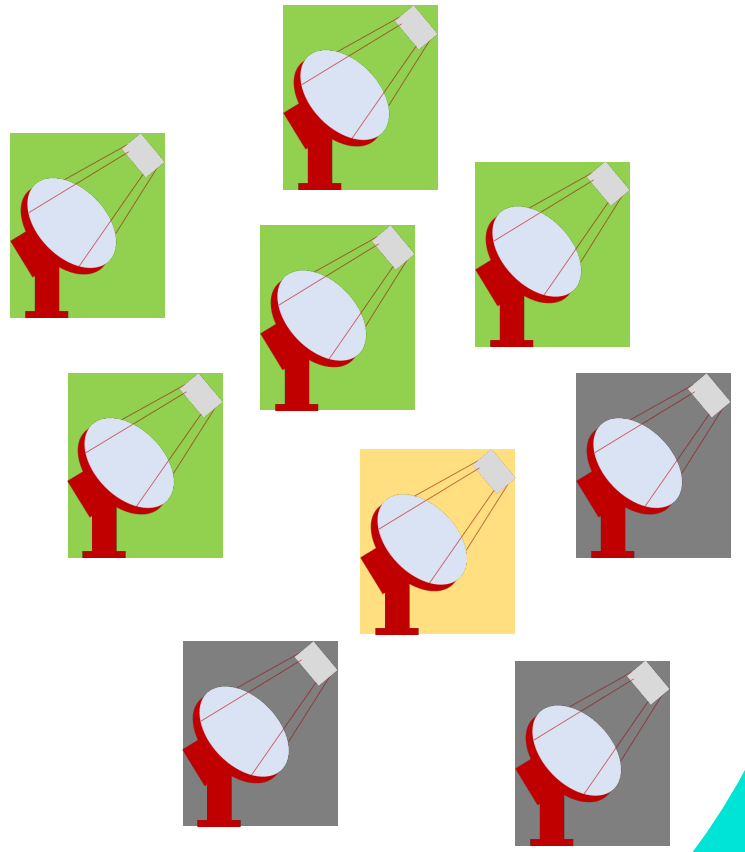


Telescopes during the night – Excersise

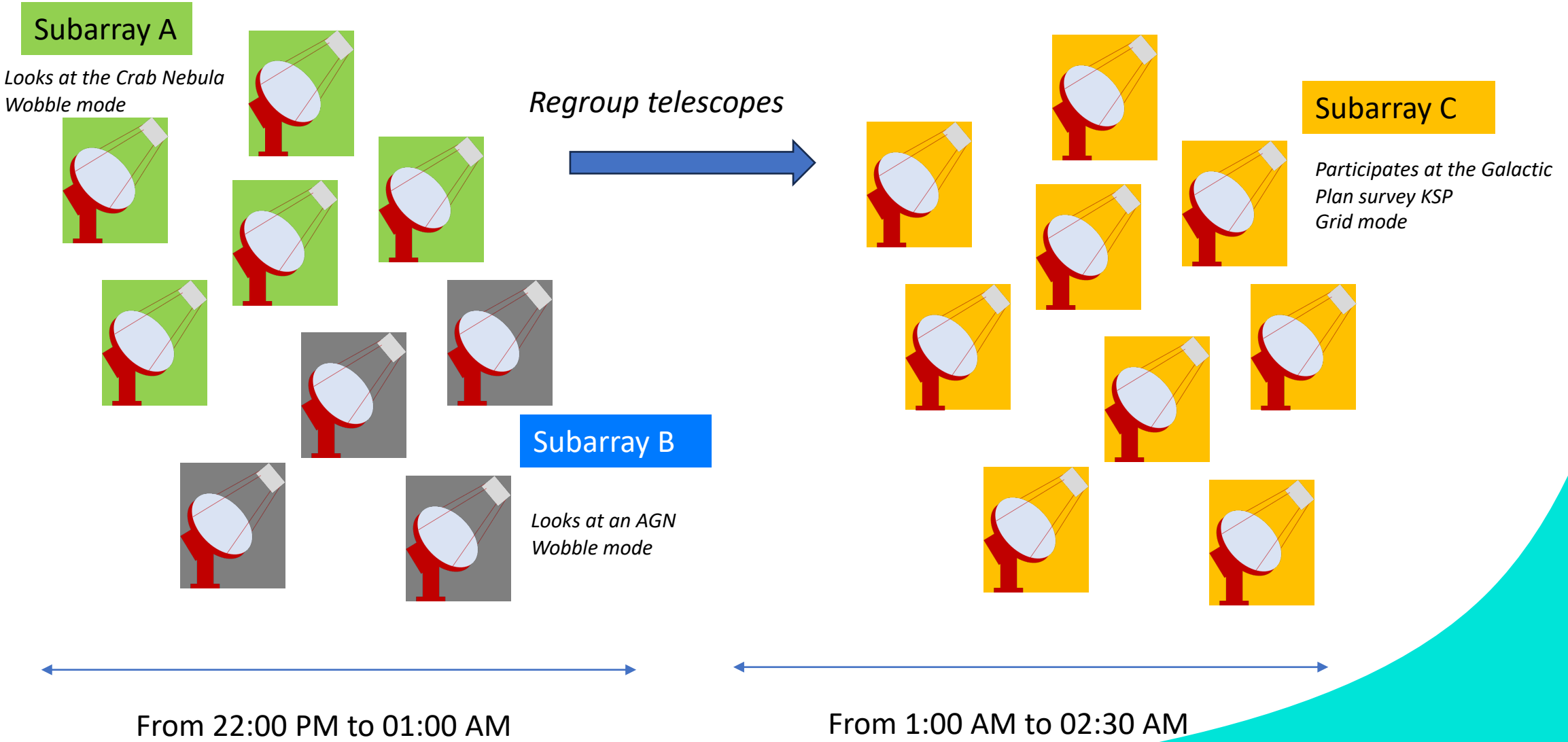
- Could you picture what are the CTAO telescopes doing during the night?

Telescopes during the night

- CTAO will support different “telescope subarray” configuration for at each site
- One possible subarray configuration will use all the telescope at the same time
- Other configurations will consist of telescope subgroups
- Telescopes operating within the same subarray point towards the same target, and use the same subarray-level trigger instance
- Allocation and operation of subarrays will be managed dynamically
- Science Alerts will cause the schedule to be adapted on the spot, and allocated subarrays
- Array common elements are operating concurrently with the telescope subarrays

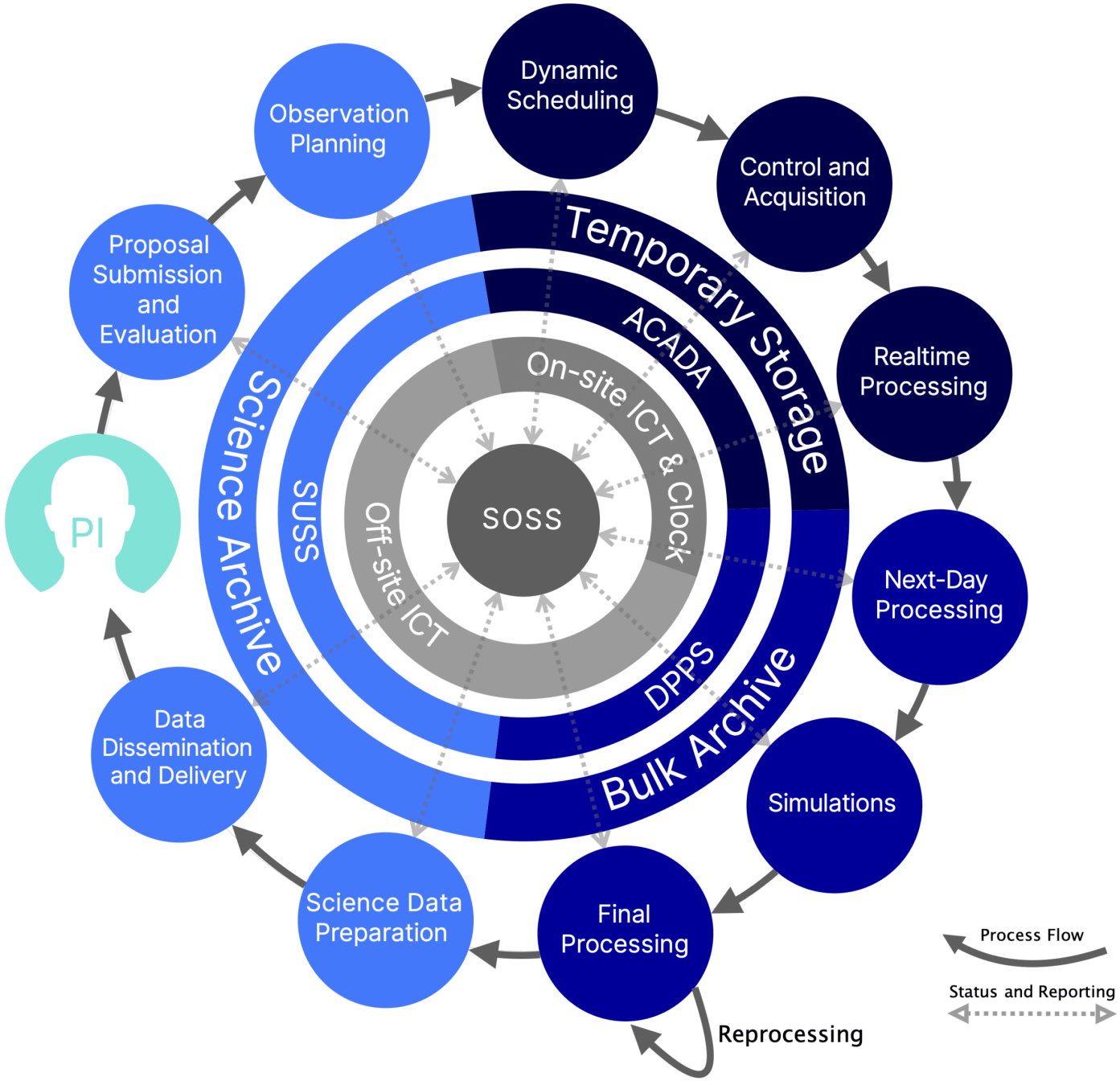


Example of Subarray Operations



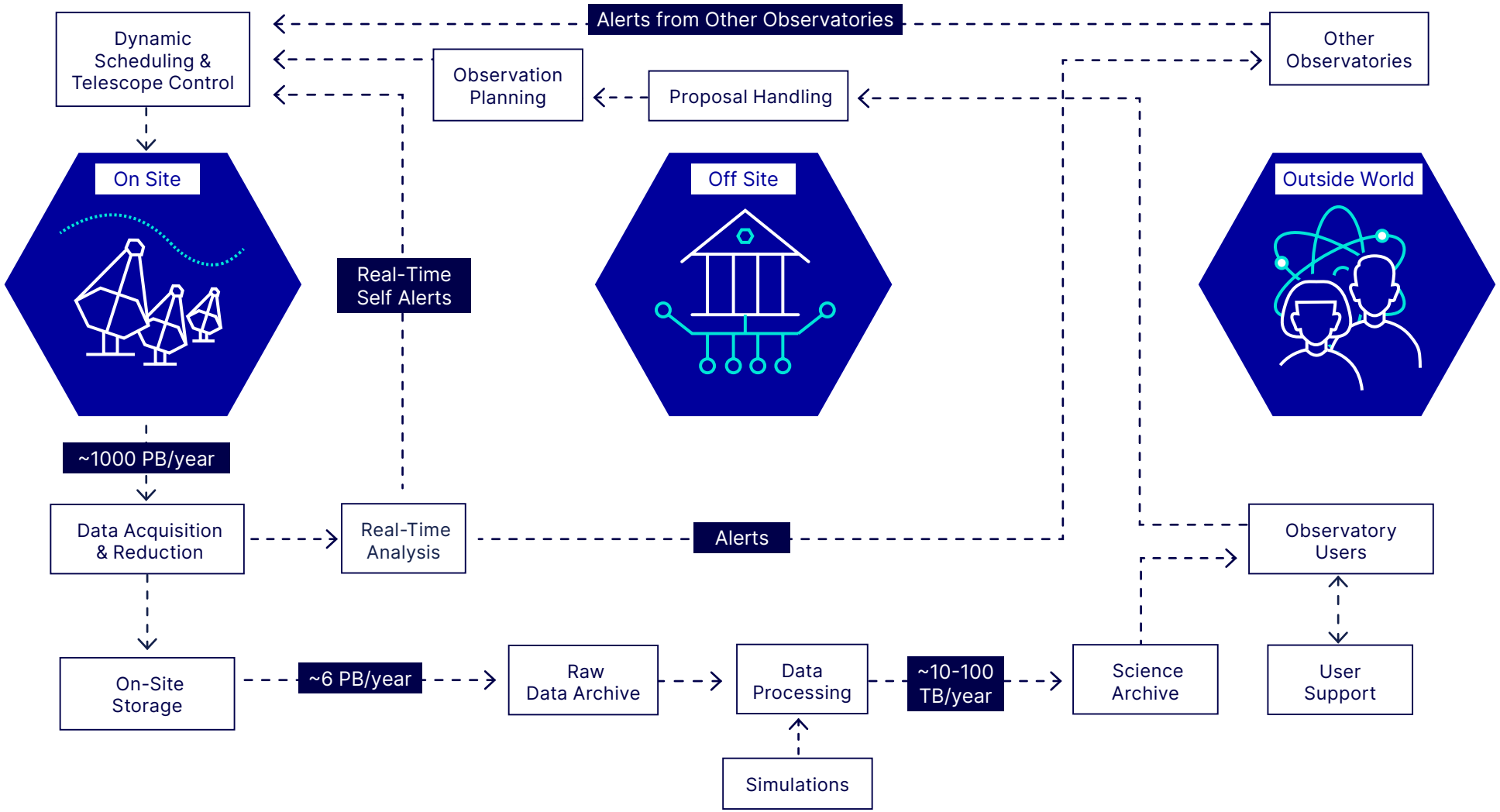
The software systems to support the CTAO Arrays

The science operations of CTAO – and the computing systems



Data flow

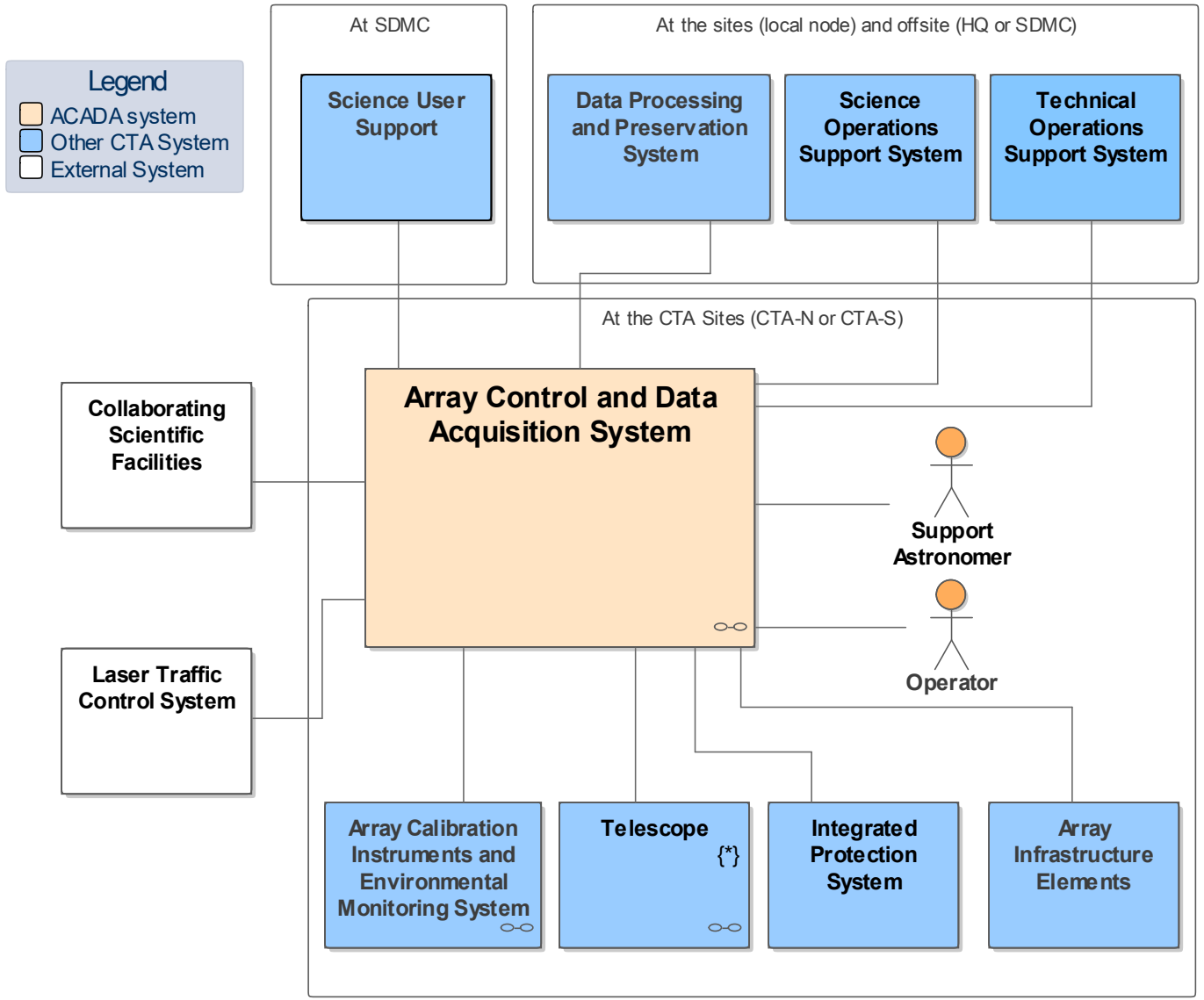
The CTAO is a BIG DATA project. The Observatory ~ 6 PB/site/year after compression.



CTAO Array Control And Data Acquisition

The software to implement Array Operations

ACADA: Role and Context



- ACADA = **A**rray **C**ontrol and **D**ata **A**cquisition system
- System for central control and data acquisition of all telescopes & instruments at both CTAO sites

How does one create an "ACADA"?

- Requirements: Express what is needed
  - Translate scientific objectives to something that can be built
  - Allocation of "budget" from higher-level requirements
  - Use previous experience by other projects
- Architecture: Define main building blocks and how things would work together
  - Has something similar done? Are there frameworks I could use? Can I define pieces of of the system that the team can crate?
  - Most importantly: is it addressing my requirements?
- Design: Details of how the building blocks of the system will be created
- Development based on releases:
  - Implement
  - Test
  - Integrate
  - Deploy
- Back to revise the design, sometimes even requirements and architecture – avoid "Waterfall"

- Team building: Who is in to participate? How will we work together?

Requirements

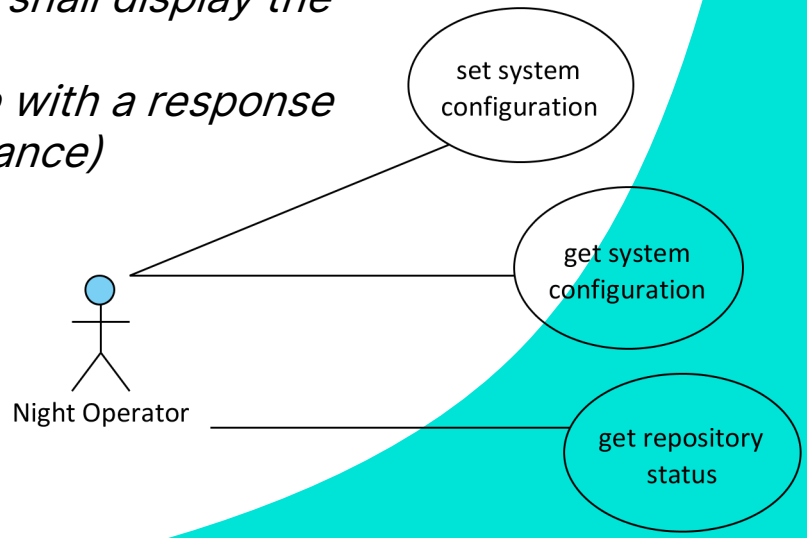
Systems engineering

- **Functional requirements:** What should be done
- **Quality requirements** (sometimes called non-functional requirements): how should be done (how fast, how secure, etc…)

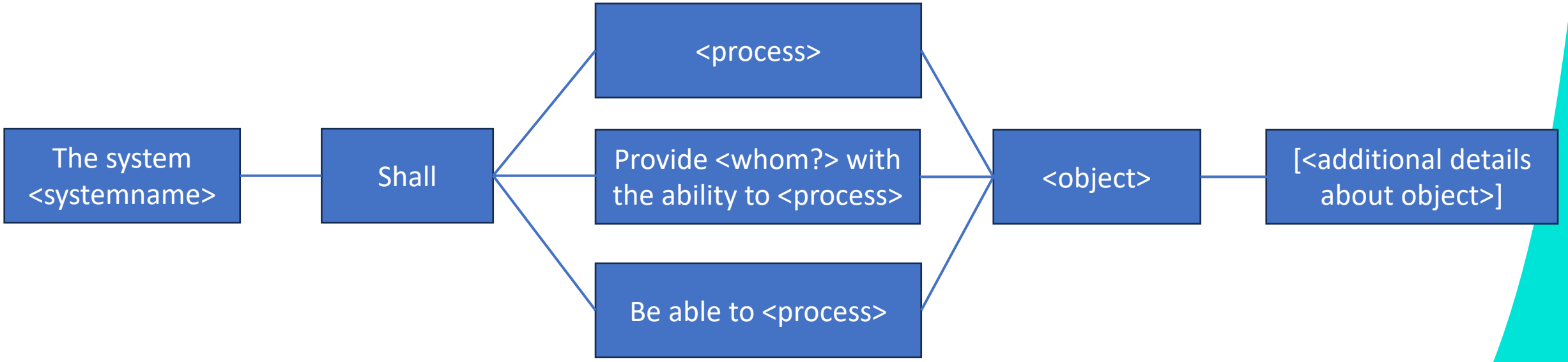
- **User requirements:** Express what the users expect they will be able to do with the system
- **Product requirements:** Defines the requirements of a particular product, including the product's purpose, features, functionality, and behavior.

- Example requirements:
  - *If the customer data entered is already present in the system, the system shall display the error message „customer already exists“. (Functional)*
  - *The system shall be able to handle 10,000 simultaneous users at any time with a response time not exceeding 3 seconds for 90% of transactions. (Quality – Performance)*

- **Use cases and user stories:** A way to express behaviour of the users with the system, and another way to express functional requirements. No time to cover them here, see “Writing Effective Use Cases, Alistair Cockburn” for more details



ACADA System requirements – excersise



- Can you imagine some requirements for ACADA?

More on requirements: https://www.sophistgroup.de/en/

ACADA requirements

- The ACADA system has more than 250 requirements associated at the product level. They are further allocated at subcomponent level, for more than 1000 requirements to realize the product. They are formulated as “ACADA shall”, or “The Operator shall be able” etc.
- Additionally, 100+ use cases describe the behaviour of the ACADA system.

ACADA Driving requirements

Wording is simplified

- Operation of eight groups of telescopes (subarrays) at the same time, dynamically formed from available telescopes.
- Support up to 99 telescopes in addition to LIDARs, stellar photometers, all-sky cameras, weather stations, ceilometers, and external light sources.
- Collect event data from all telescope cameras.

Data throughput	24 Gb/s/LST (up to four)
	12 Gb/s/MST (up to 25)
	2 Gb/s/SST (up to 70)
Individual telescope trigger rates	15 kHz/LST (up to four)
	14 kHz/MST (up to 25)
	1.2 kHz/SST (up to 70)
Array-level trigger rates	40 kHz
Dead time after a coincidence	of 250 ns
DVR factor	Up to 50

- Reduce the data volume of the Cherenkov telescope camera on the fly, up to a factor of 50.
- Acquire all hardware-related monitoring points provided by the Array Elements (AEs), up to an overall maximum of 200,000 data points.
- Handle errors and alarms of ACADA and AEs, suggesting to the CTAO operator possible mitigation actions to be taken.
- Interface international networks for astrophysical transients; receiving, processing and filtering incoming information, triggering observations within 5 seconds of the receipt of the external alert.

ACADA Driving requirements

Wording is simplified

- Provide preliminary data reduction and analysis and conduct a full field-of-view search for transient and/or time-variable phenomena from those data.
- Automatic generation, and subsequent revision of the observational schedule by ACADA during an observing night without the need for human intervention
- Capability for a full restart of ACADA software within 2 minutes, with full monitoring and alarm functionality of all available AEs available within 5 minutes.
- Automatic operation throughout the observing night in the absence of alarms.
- Suitability for being operated by two single individuals: an operator and one support astronomer located in a control room.
- The core functionality of ACADA, needed for system control and the safe execution of observations, is to be available during 99% of the time in which observations are possible.
- Be deployable in any standard computing node running a Red Hat Linux derivative in the onsite data centre (the current valid version being Almalinux 9) and does not require specially tailored machines.

Designing software for a distributed system

- A control system based on a single computer is only possible in a small setups
  - Limited CPU, memory, storage, and network bandwidth.
  - Single point of failure → if it crashes, everything stops.
  - Limited scalability → you can only “scale up” by upgrading the hardware, which has physical and financial limits.
- Distributed system
  - A collection of interconnected computers, servers, or devices that work together as a single, cohesive unit to achieve a common goal.
  - The components communicate and share resources with one another to function efficiently and effectively.
  - They can be in proximity, such as within a single building, or spread across vast distances, connecting over the internet.

Designing software for a distributed system

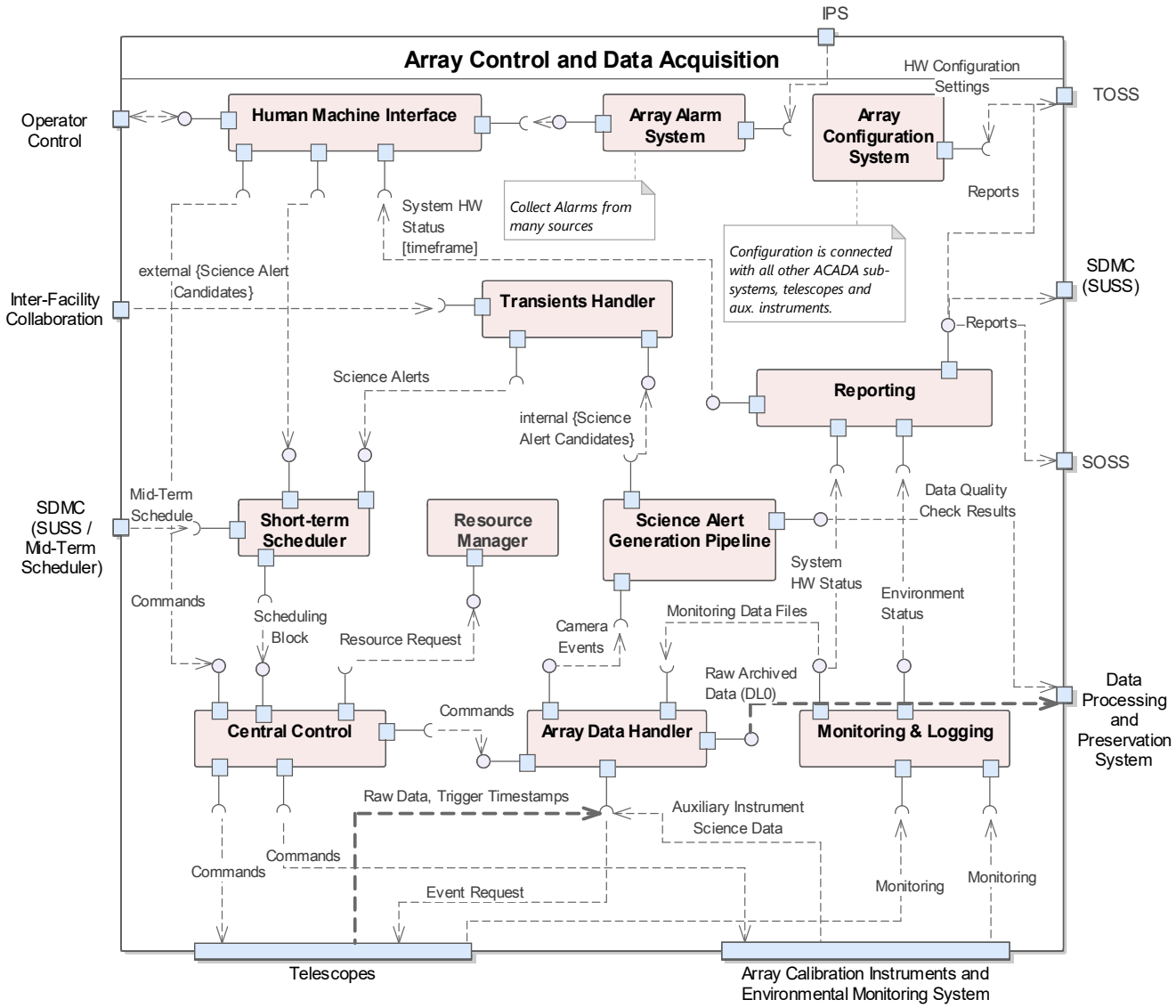
- Distributed system paradigms
  - Message-exchange pattern
  - Publish-Subscribe Message Model (ex:, RabbitMQ, ActiveMQ, Kafka)
  - Request-reply
  - Remote Procedure Call (RPC)
  - Distributed Objects
    - Application Objects distributed over the network
    - Examples: Enterprise Java Beans, Microsoft DCOM, Java RMI, ZeroC ICE, CORBA

Control frameworks

- A distributed telescope array control system has an “industrial” equivalent, and it is called ”Industrial Control System” (ICS). Subtypes exist:
  - Supervisory control and data acquisition (SCADA) systems, or Distributed Control Systems (DCSs)
- The scientific community has developer over the years common solutions for ICSs as form of ”frameworks”. The most know ones:
- EPICS - Experimental Physics and Industrial Control System: https://epics-controls.org/
- TANGO - TAco Next Generation Objects: https://www.tango-controls.org/
- ACS – ALMA Common Software: https://confluence.alma.cl/display/ICTACS/ICT+ALMA+Common+Software

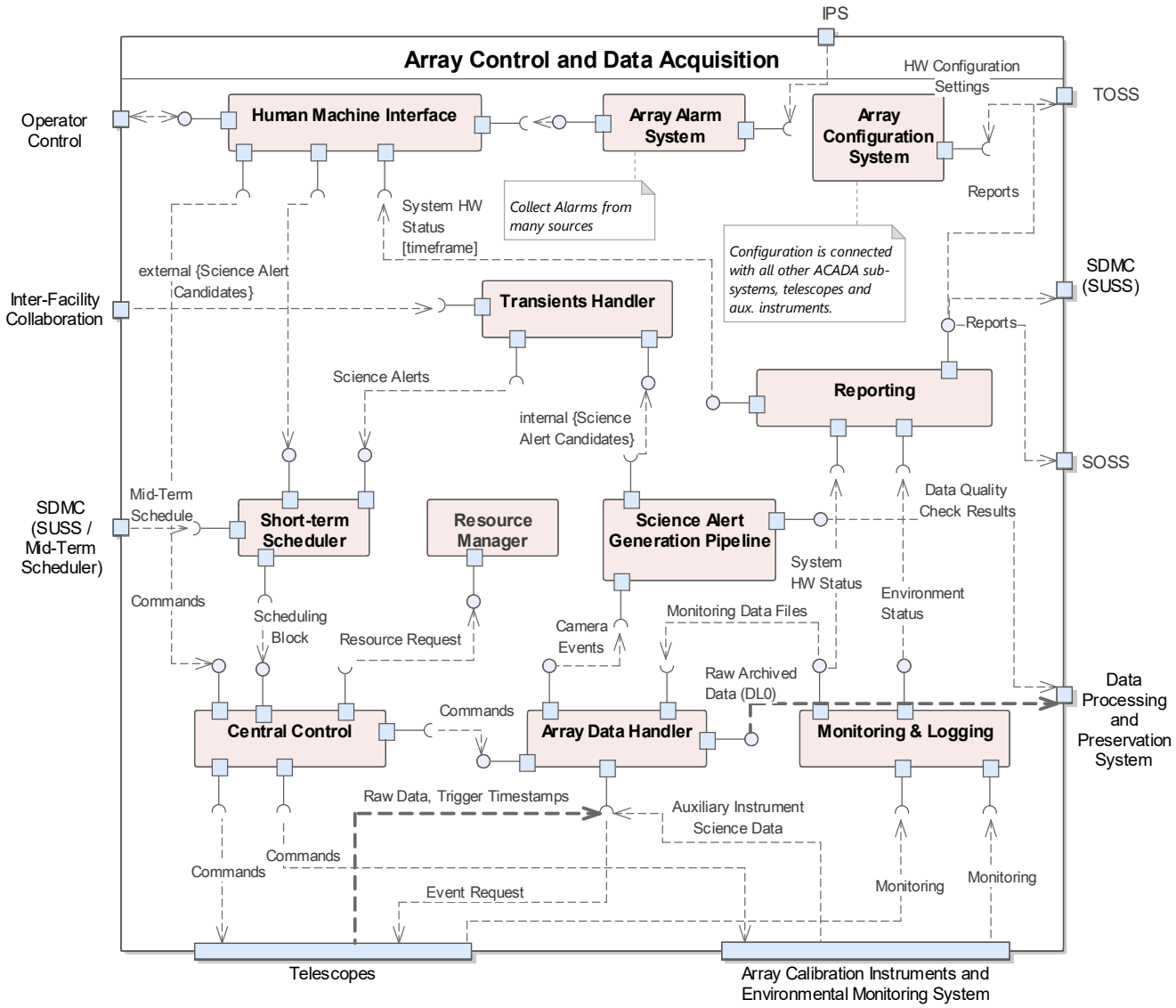
- CTAO uses ACS as the basic ingredient for control of the systems, which used a mixture of the Distributed system paradigms

ACADA System architecture



- The ACADA Architecture is derived following model-based systems engineering practices and details how the system is decomposed in building blocks and how they work together.
- Rationale for design choices are documented.

ACADA System architecture

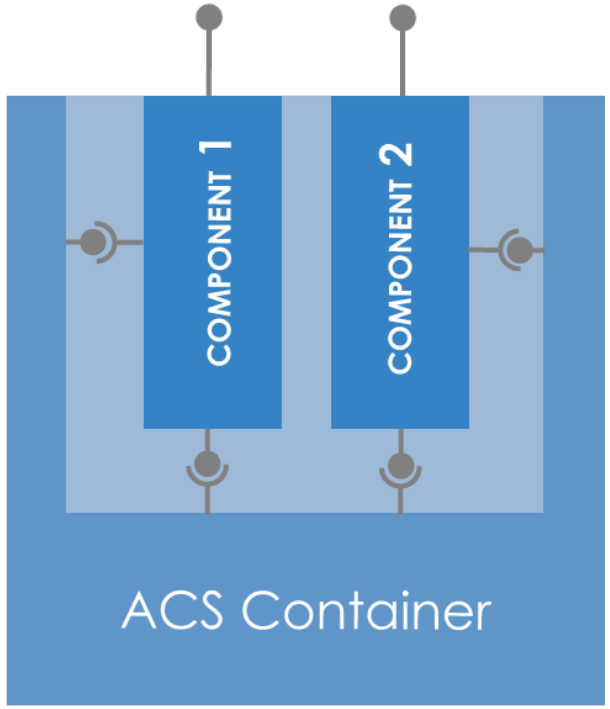


Tech Stack

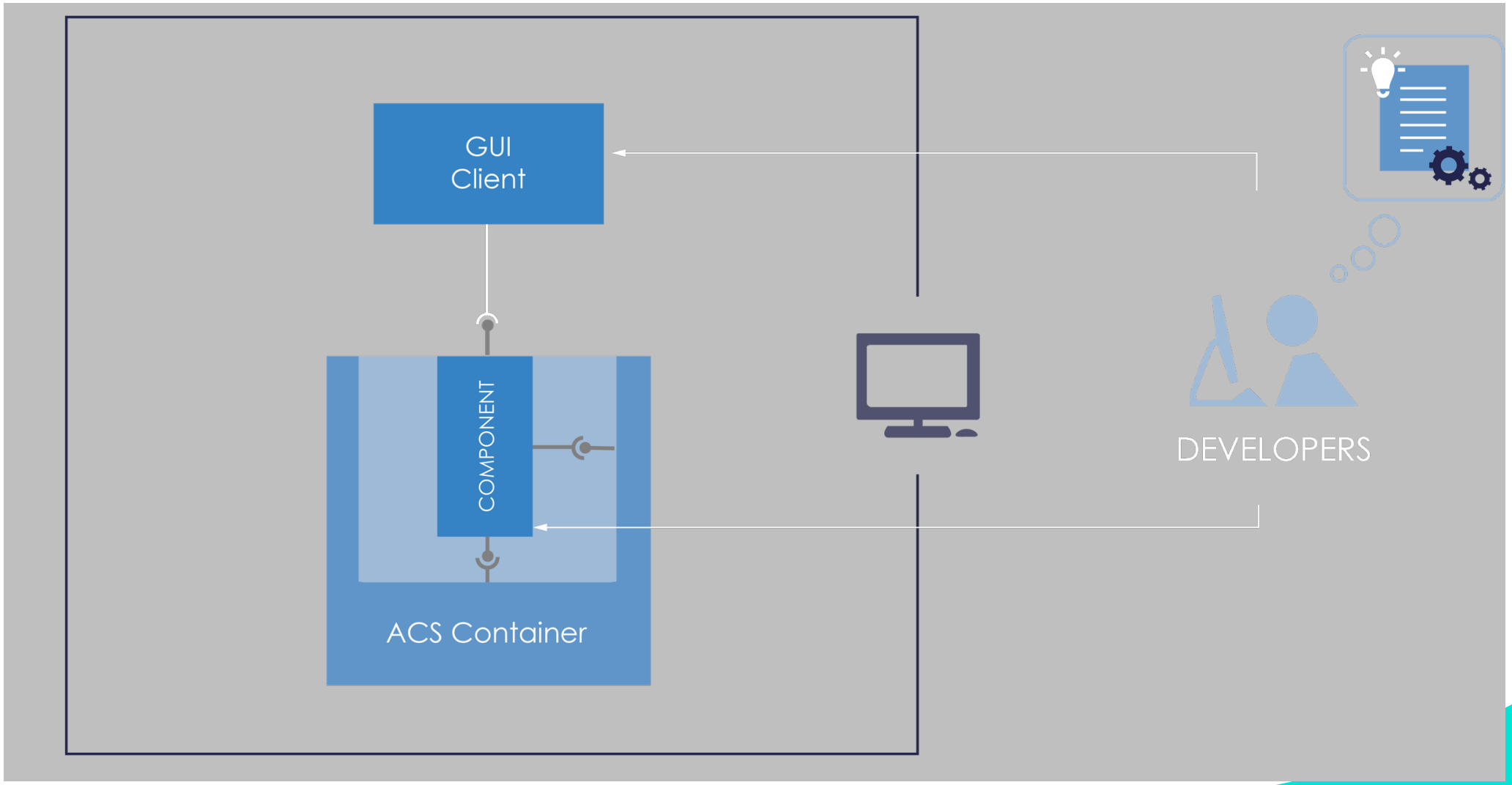
- **Middleware:** Alma Common Software (ACS), OPC UA (for access to monitoring data only)
- **Programming languages:** Python, Java, C++, Javascript (for the Human-Machine Interface (HMI) front-end only)
- **Databases:** MySQL, MongoDB, Redis, Cassandra
- **Messaging and serialization:** Apache Kafka, ZeroMQ, Google Protocol Buffers. CORBA (via ACS), REST/JSON
- **Workload management system:** Slurm
- **Containers:** Docker
- **Other:** ESO's Integrated Alarm System(IAS)

Creating an ACS application (I)

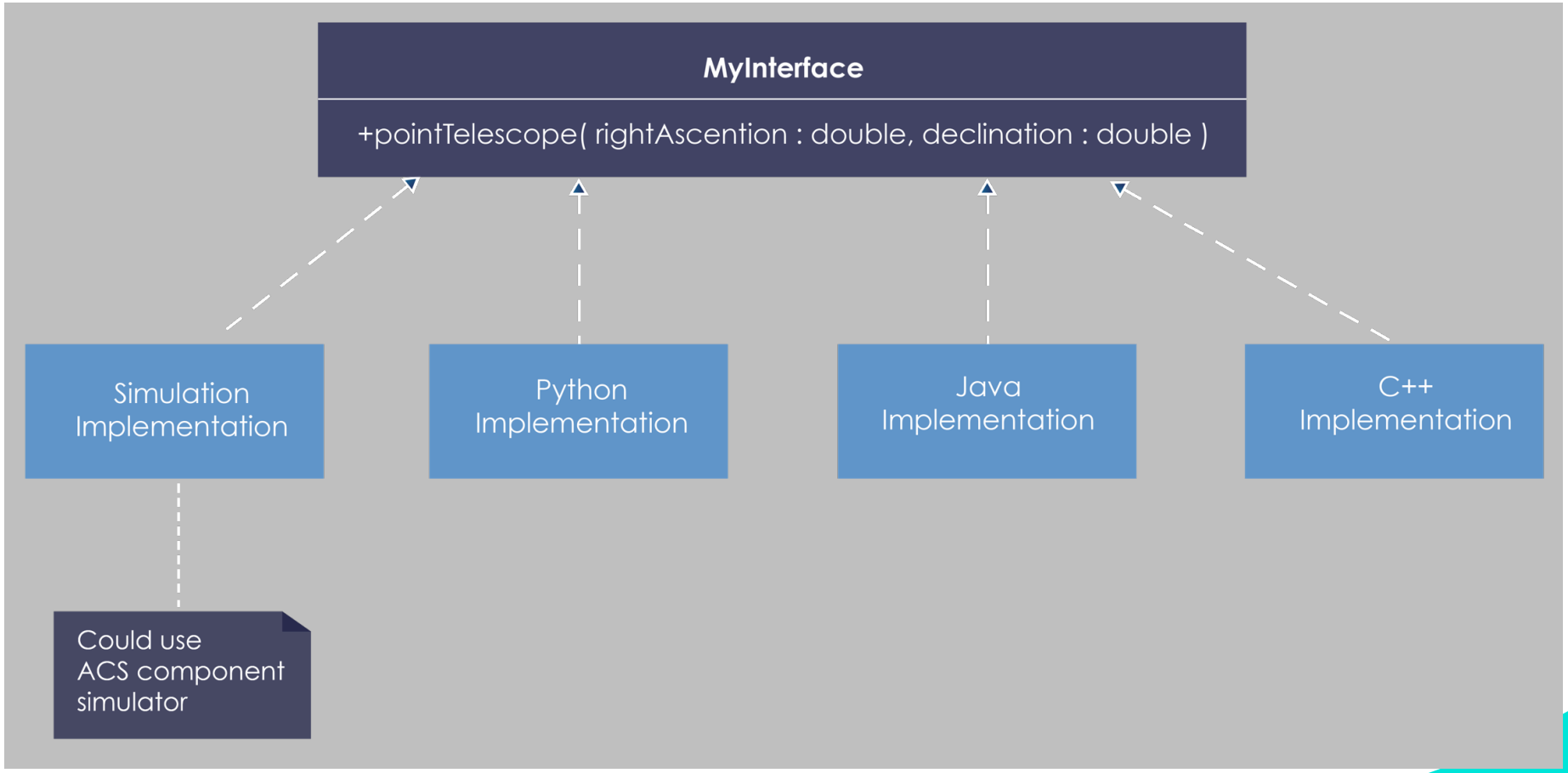
- An ACS “system” is a a collection of
  - **Components**: Entities that provide some functionalities, for example a component can represent a telescope, an instrument, a data streaming service…
    - Developers mainly deal with implementing components.
    - Components can communicate with other components.
    - Components cannot run by themselves.
  - **Containers**: Runtime environment from Components. Components are deployed in containers. Containers can be configured, normally by those setting up the system deployment
  - **Clients**: Interfaces to components, e.g., HMIs, data capturing processes etc.
    - Implemented by developers
  - **Services**: Infrastructure provided by ACS, configurable by those setting up the system.



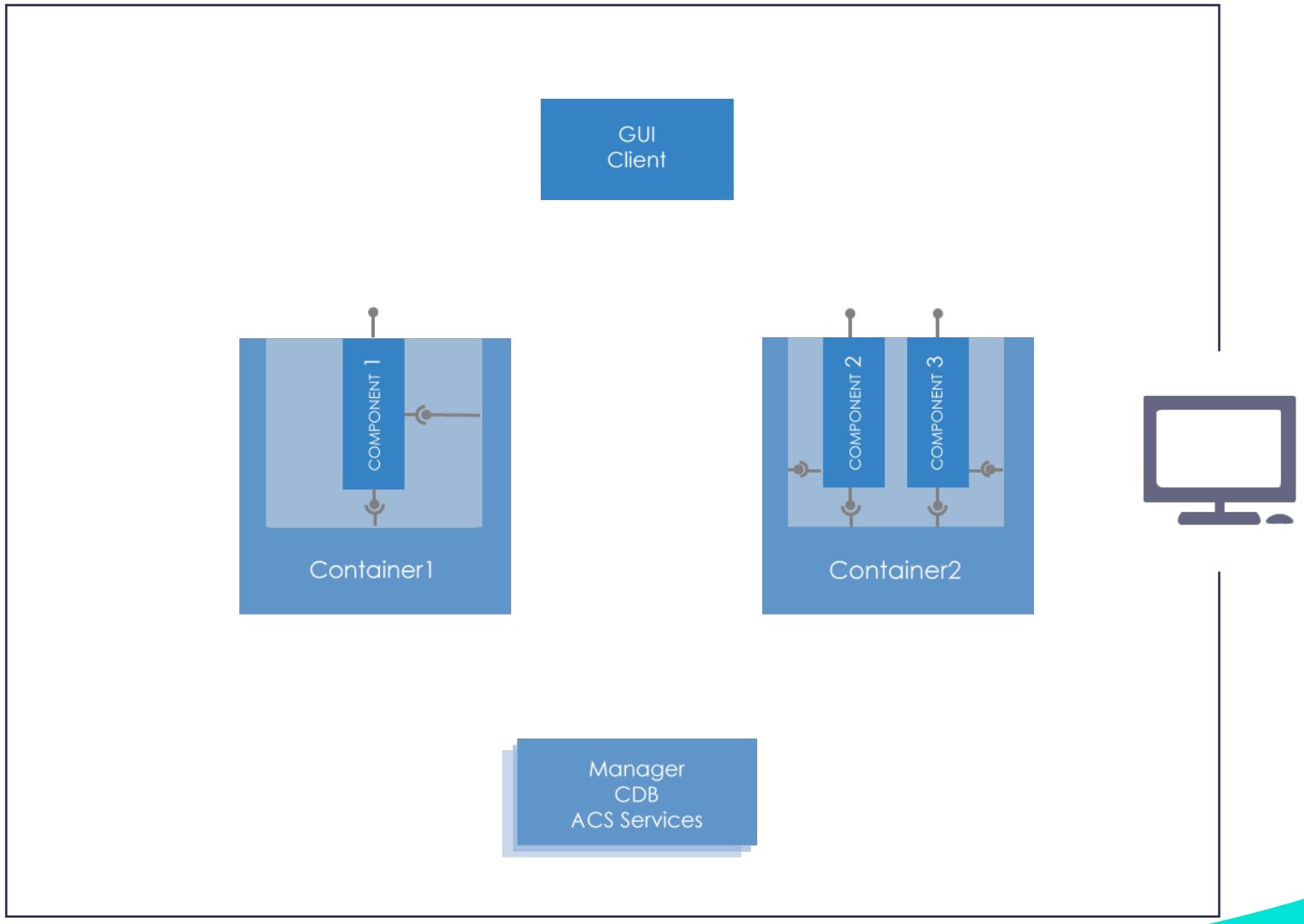
Creating an ACS application (I)



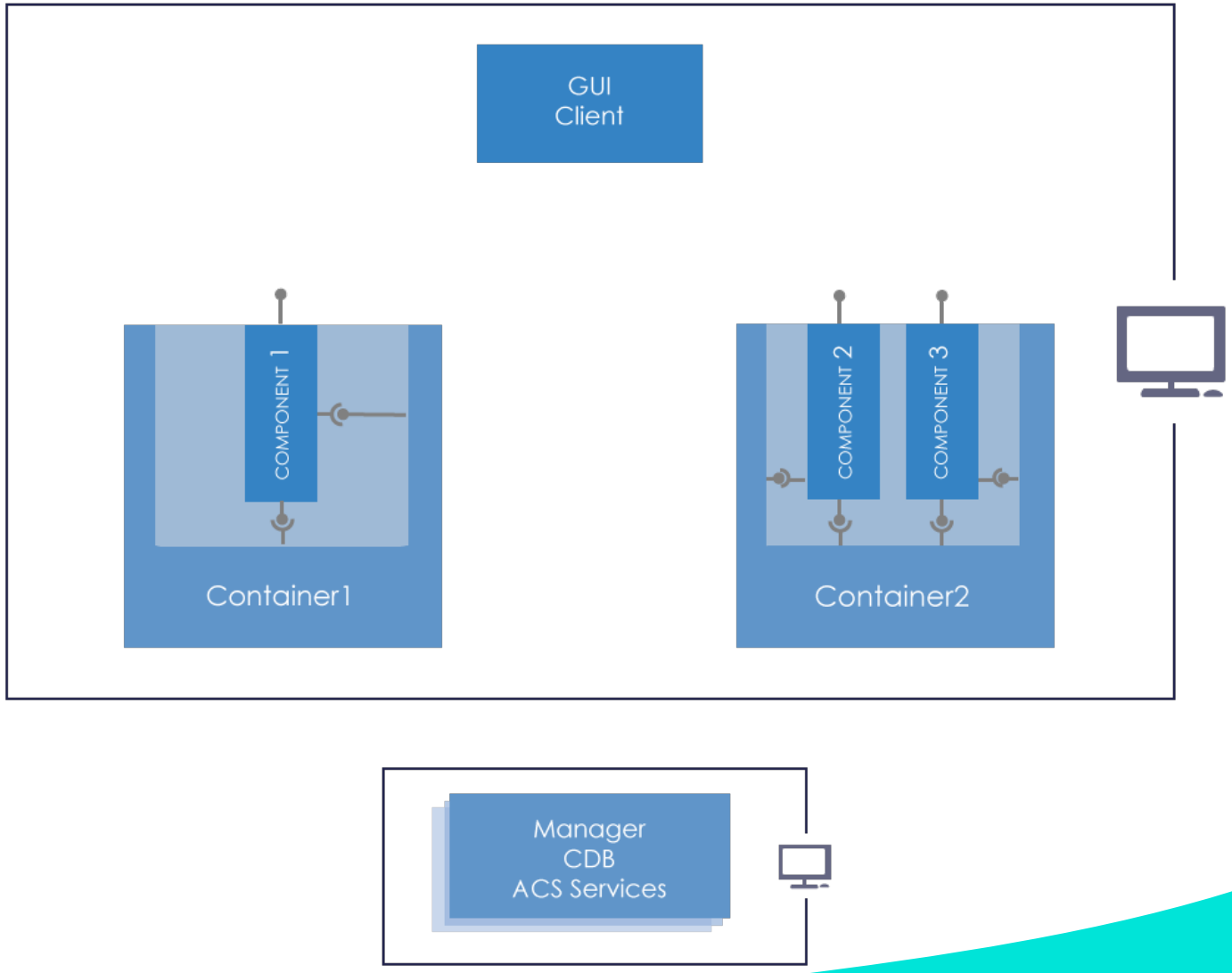
Creating an ACS application (I)



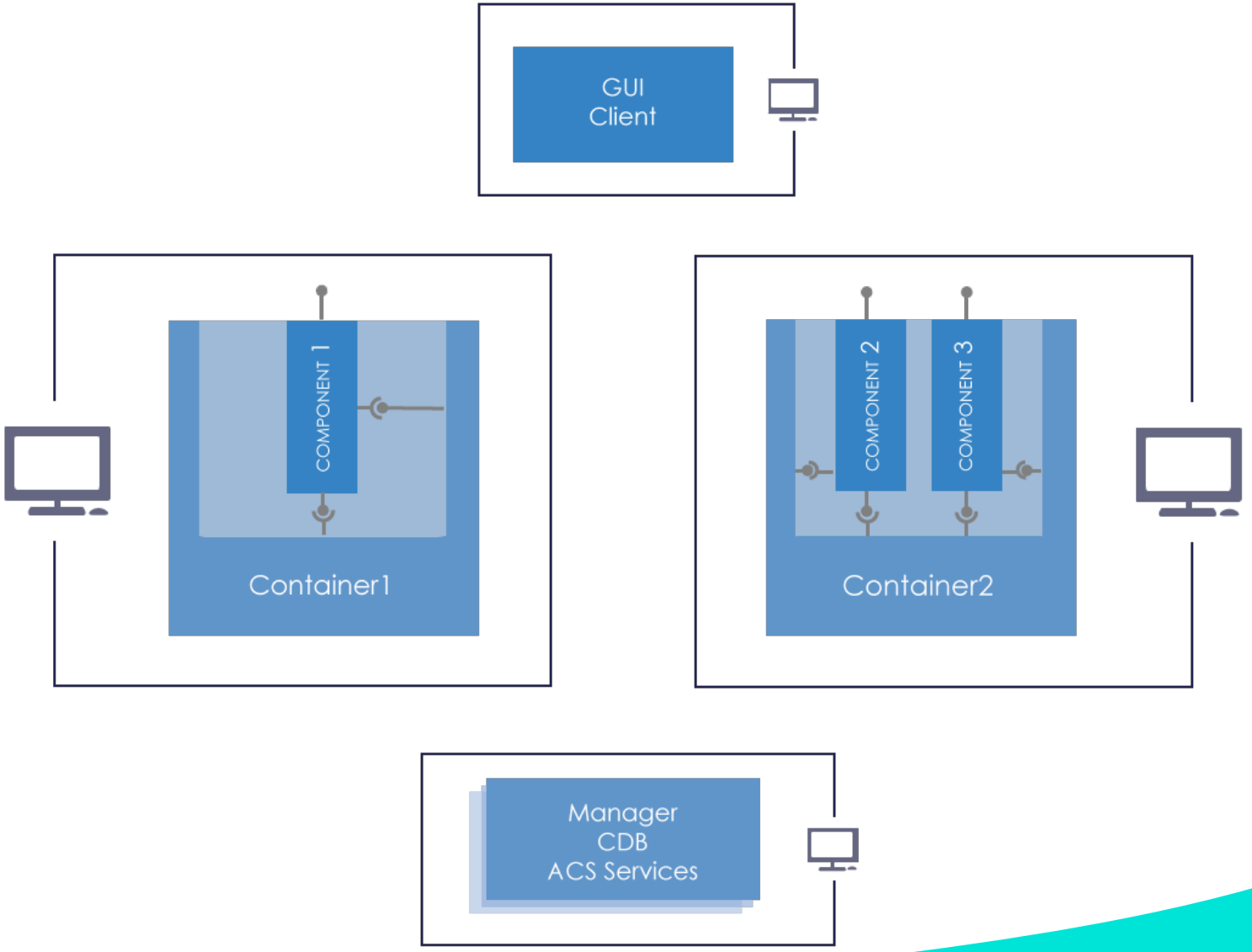
Deploying an ACS application (I)



Deploying an ACS application (II)



Deploying an ACS application (III)

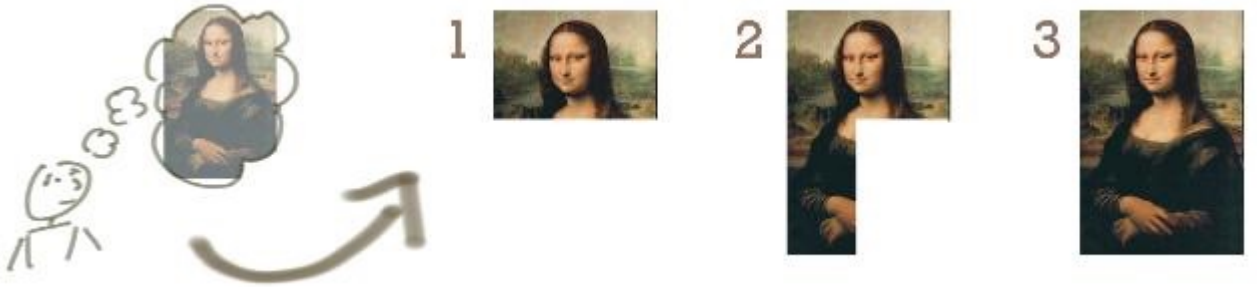


Creating an ACS application – for more information

- For more on ACS, you can follow the training information of the latest workshop: 13th ACS Workshop - 7th, 8th and 9th of May 2025, hosted at the CTAO SDMC https://confluence.alma.cl/pages/viewpage.action?pageId=216042088

Incremental and Iterative Software Development life-Cycle (SDLC)

Incremental development:

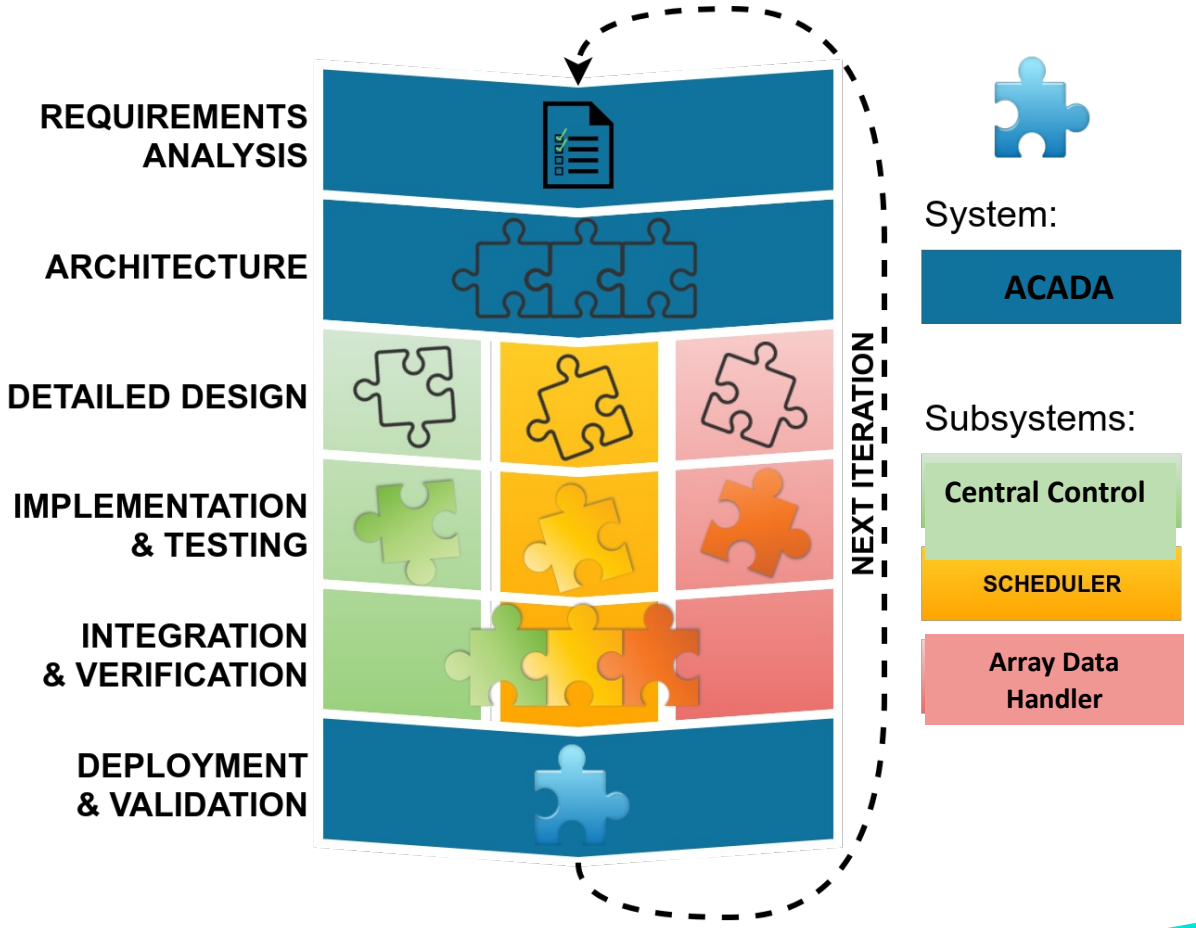


Iterative development:

Source: Jeff Patton
https://jpattonassociates.com/dont_know_what_i_want/

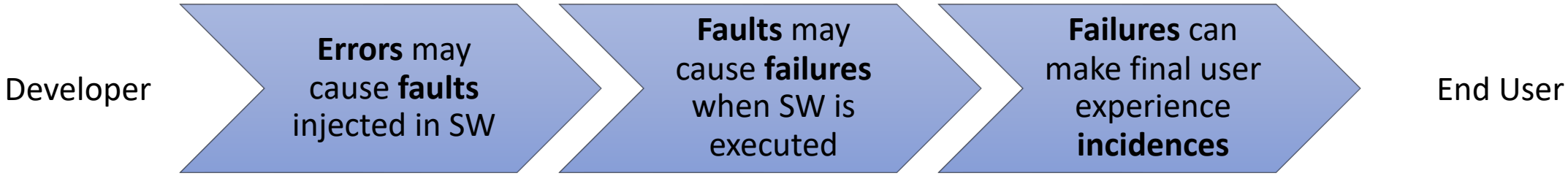
Software Development Life-Cycle – Iterative+Incremental

Inside every release...



Software quality

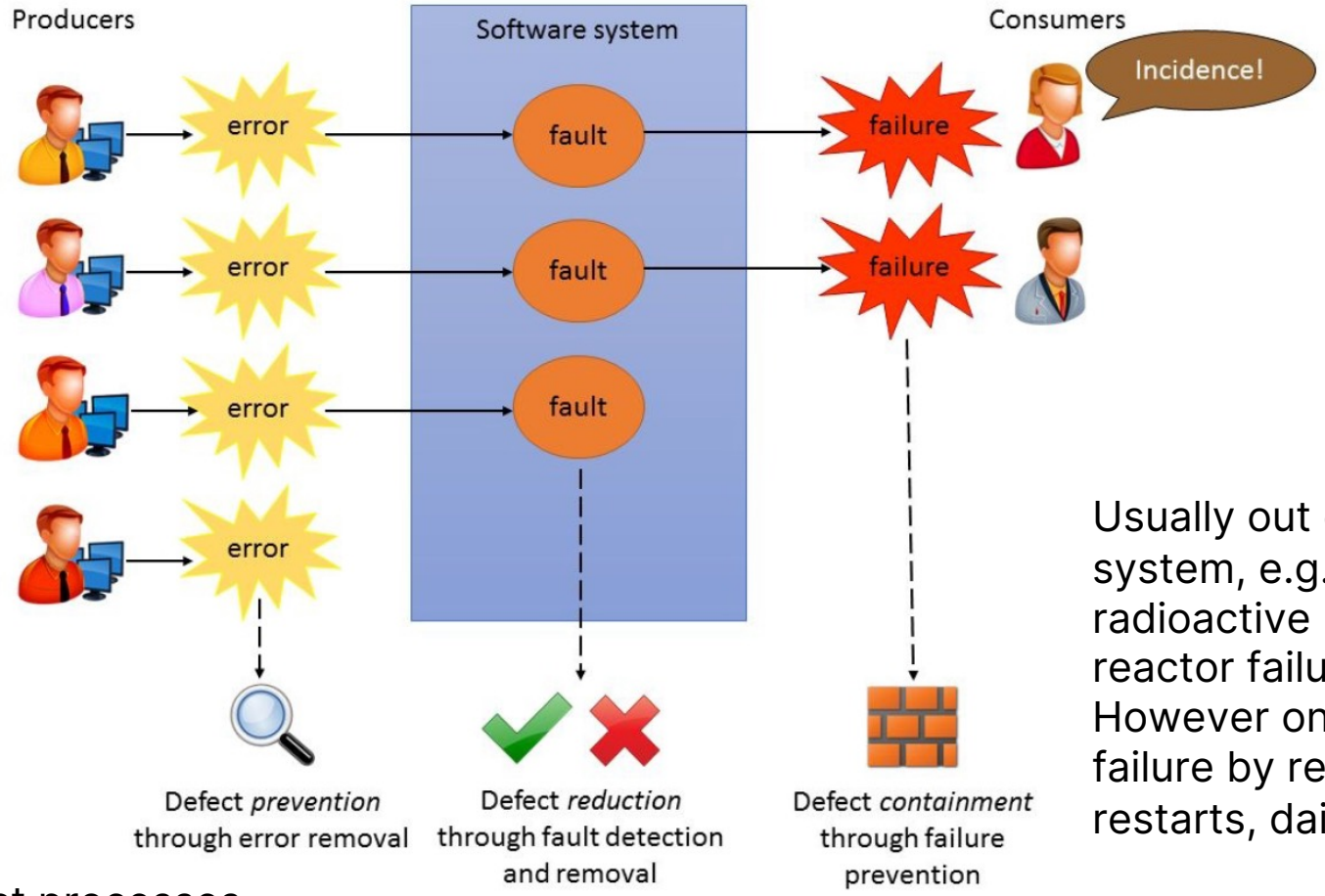
- Quality assurance (QA) Systematic, planned set of actions necessary to provide adequate confidence that the software development and maintenance process of a software system product conforms to established specification as well as with the managerial requirements of keeping the schedule and operating within the budgetary confines. [Software Quality Assurance, Daniel Galin(2004)]



Software quality

- Many of you probably create software. How do you make sure that the software is good?

Software quality



E.g. correct processes, expertise, programming standards, IDEs.

E.g. CI, testing and static analysis

Usually out of Scope of SW system, e.g. walls to contain radioactive material in case of reactor failures
However one can mitigate failure by redundancies, quick restarts, daily reboots etc.

Software quality In CTAO

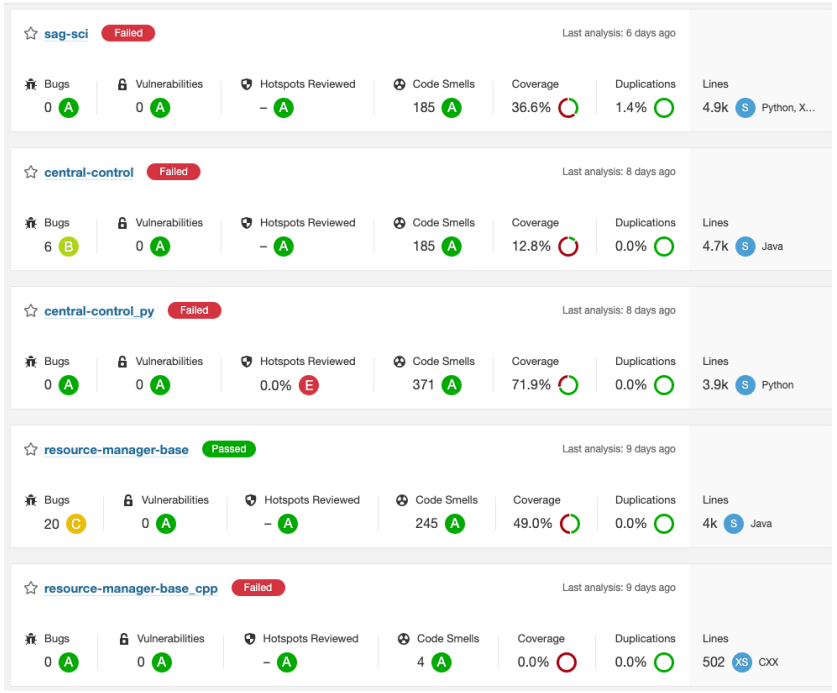
QA and CI/CD

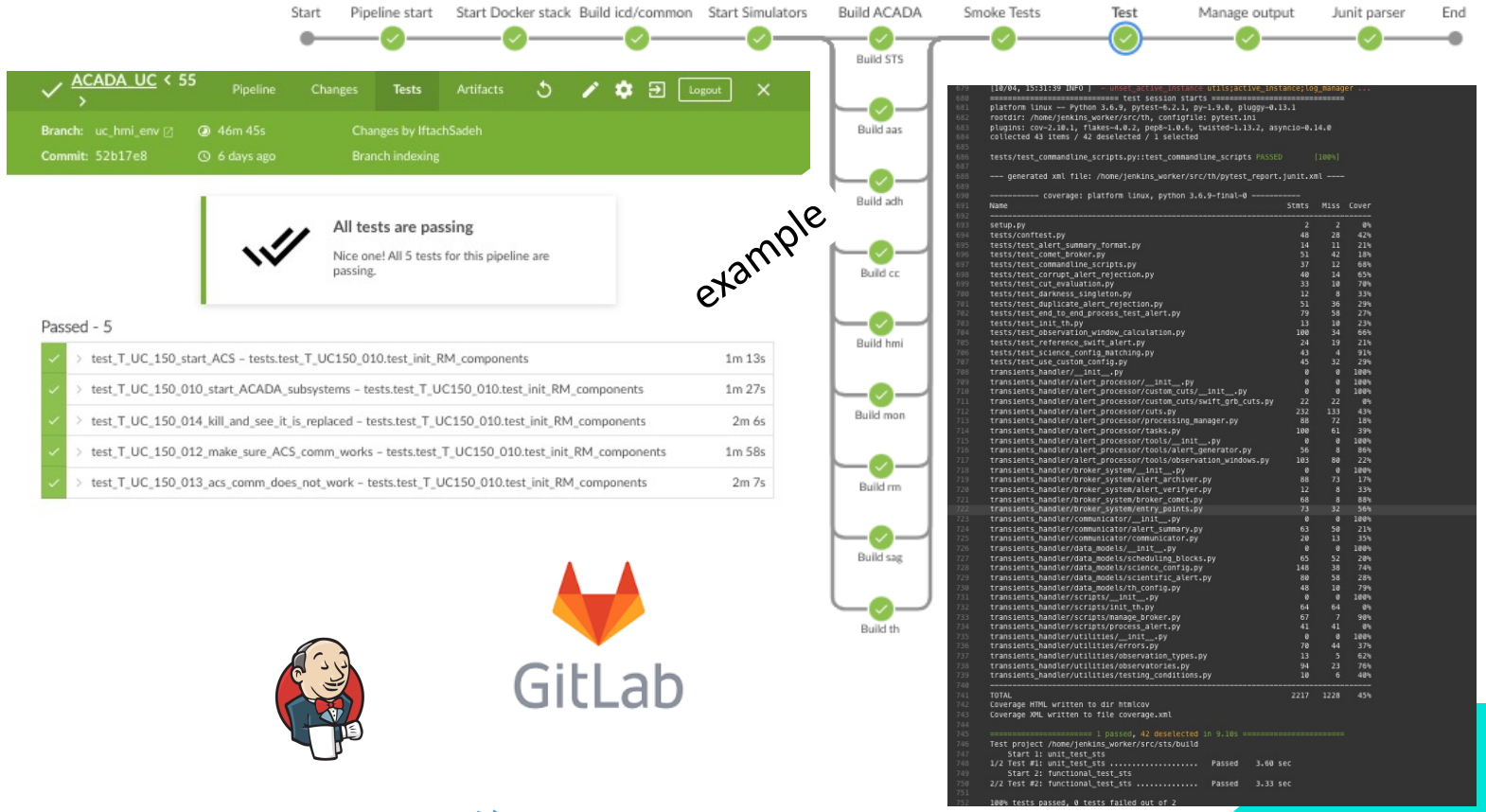
- In CTAO we address the QA issue with procedures and tools
- Procedures: continuous integration (CI) and continuous delivery (CD) – CI/CD, adhesion to standards, and peer code reviews
- CI/CD
  - Tooling: GitLab CI and Jenkins.
  - We make sure that submitted code compiles, test are passed and can be be deployed. Measure how much code have been tested (coverage)
- Static code analysis:
  - We inspect the software with tool to check for possible issues, naming convention violation, code repetitions etc. We define “quality” gates for a piece of software be acceptable.
  - Tooling: SonarQube
- High quality code will make the software easier to maintain and avoids issues during operations or system upgrades, thus, helps with operation costs and KPIs…

… But it also takes more effort (increase construction costs).

Software quality

QA and CI/CD





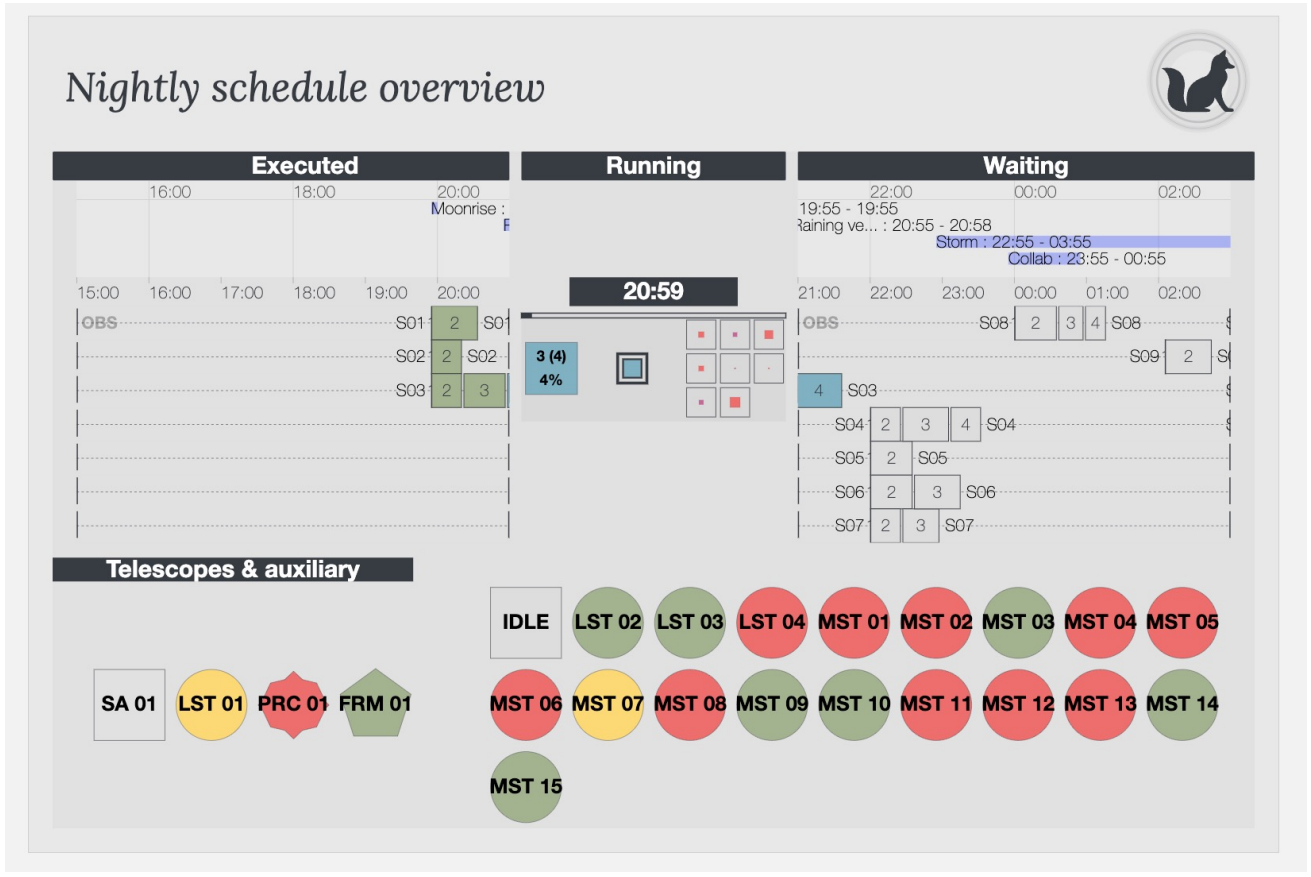
Design of HMIs

- Human Machine Interfaces (HMIs) are the main instrument how the operators of a system can interact with it.
- For large and complex systems, a good HMI design is fundamental for optimizing the role of the operators and for the proper functioning of the system.
- Aspects into account:
  - Visual perception, short-term memory, long-term memory, problem solving and reasoning, decision making, learning, …
  - Minimize work for operator: Allow unnecessary manual operation, clicks, moving from one screen to anther etc.
  - Minimize stress level: Only draw attention of operators when really needed
  - Do not overload with information, only convey the necessary information
- Input from operators during the design is very beneficial.

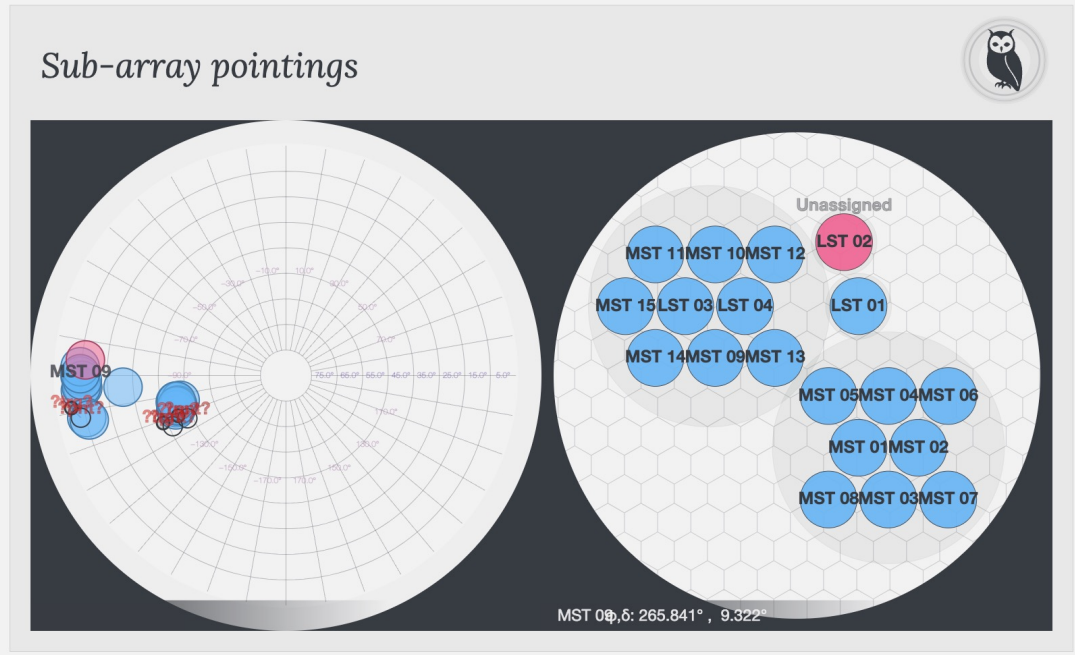
Design of HMIs - exercise

- What do you think is important to visualize in an IACT control room?

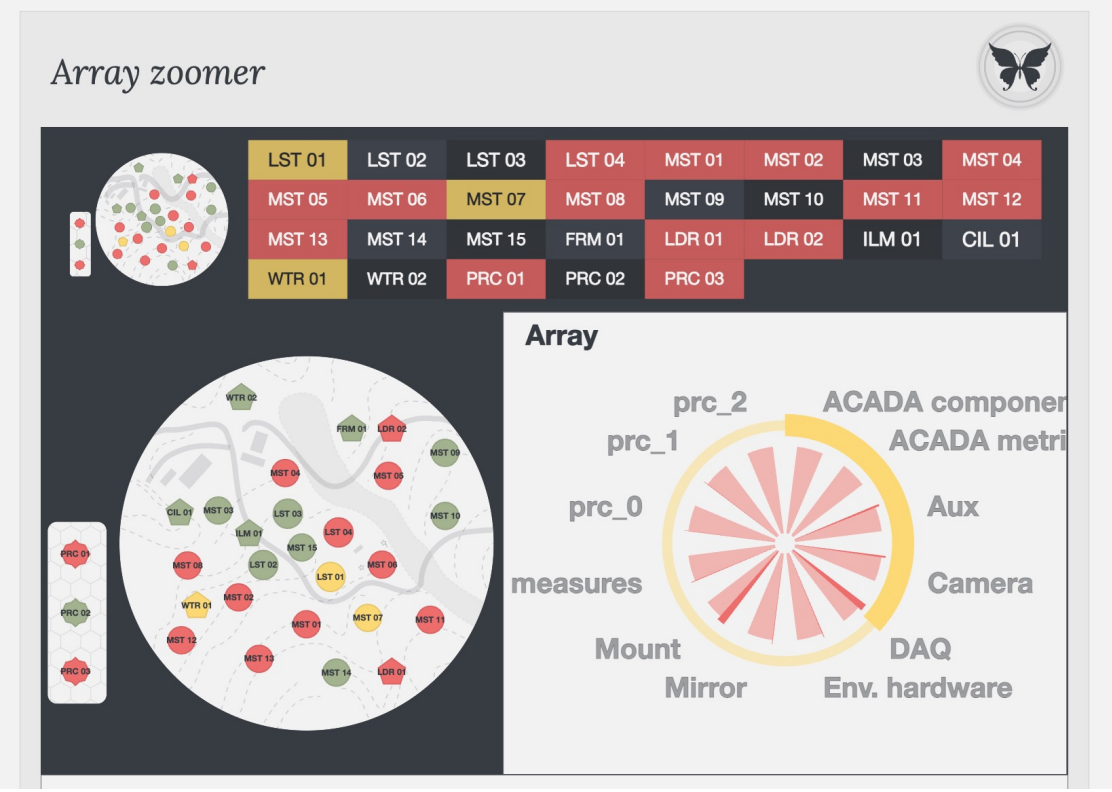
Some CTAO ACADA HMI concepts

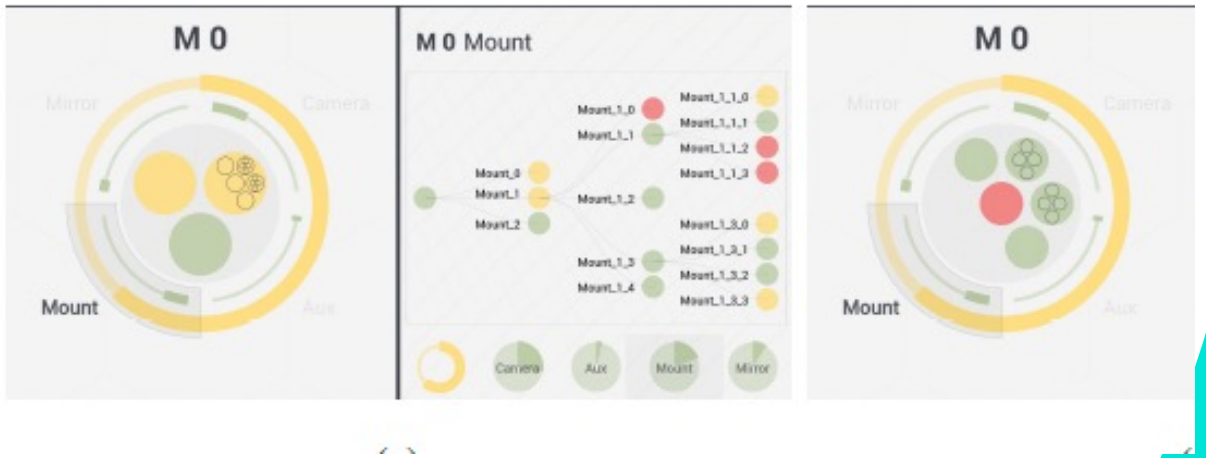


Credit: Iftach Sadeh and ACADA HMI team



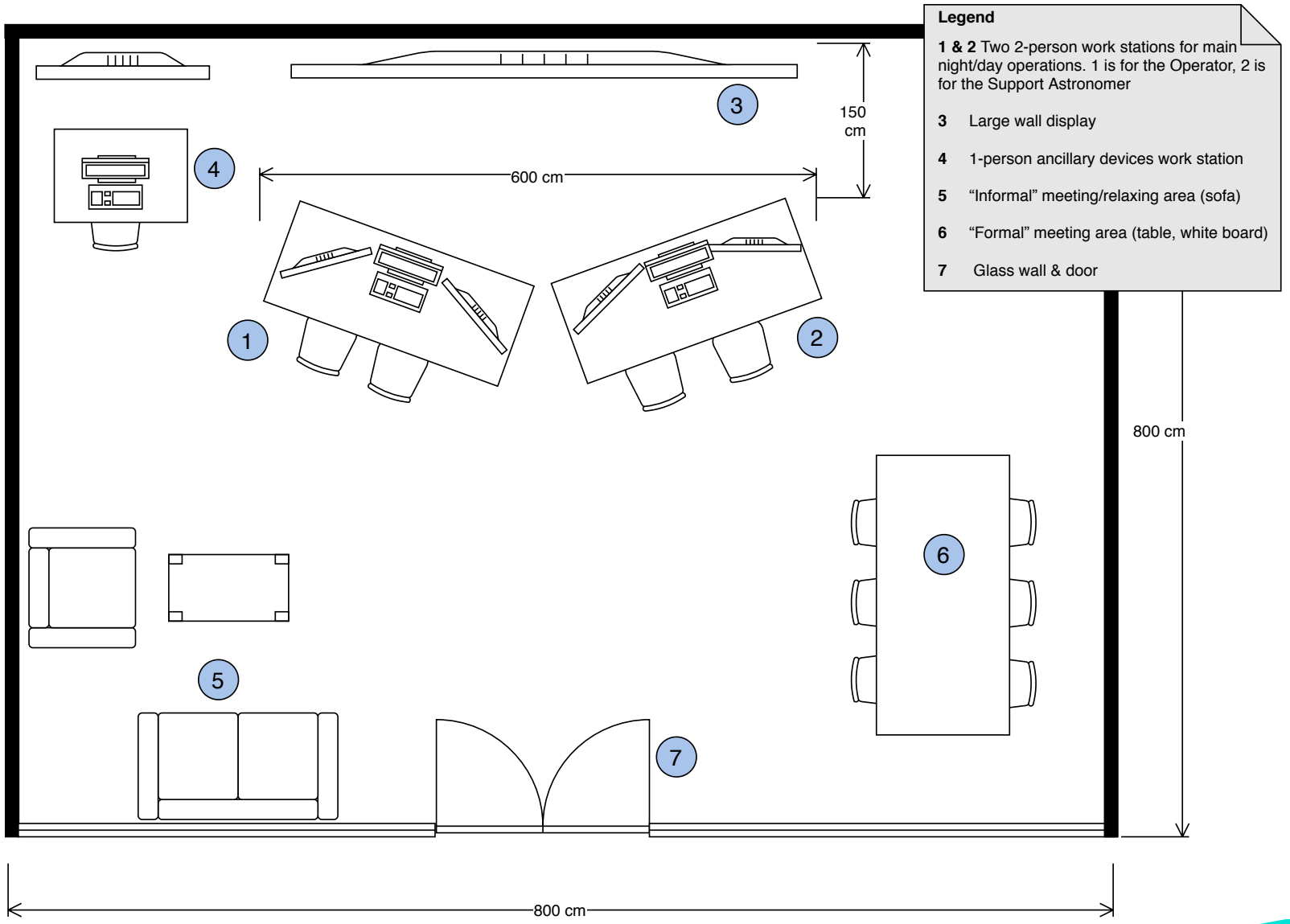
Some CTAO ACADA HMI concepts





Credit: Iftach Sadeh and ACADA HMI team

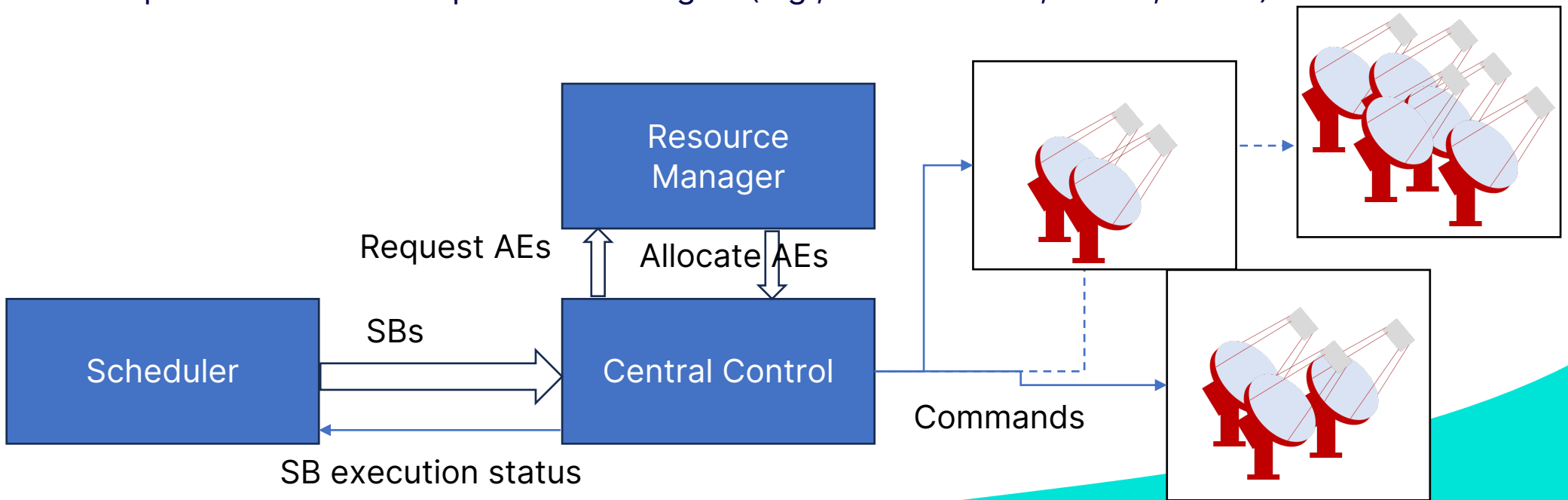
Control room



- Operator: Technically-oriented panels
- Support astronomer: Scientific-oriented panels
- Common wall panels to communicate (e.g., weather, scheduling blocks)

Scheduling, control and supervision aspects

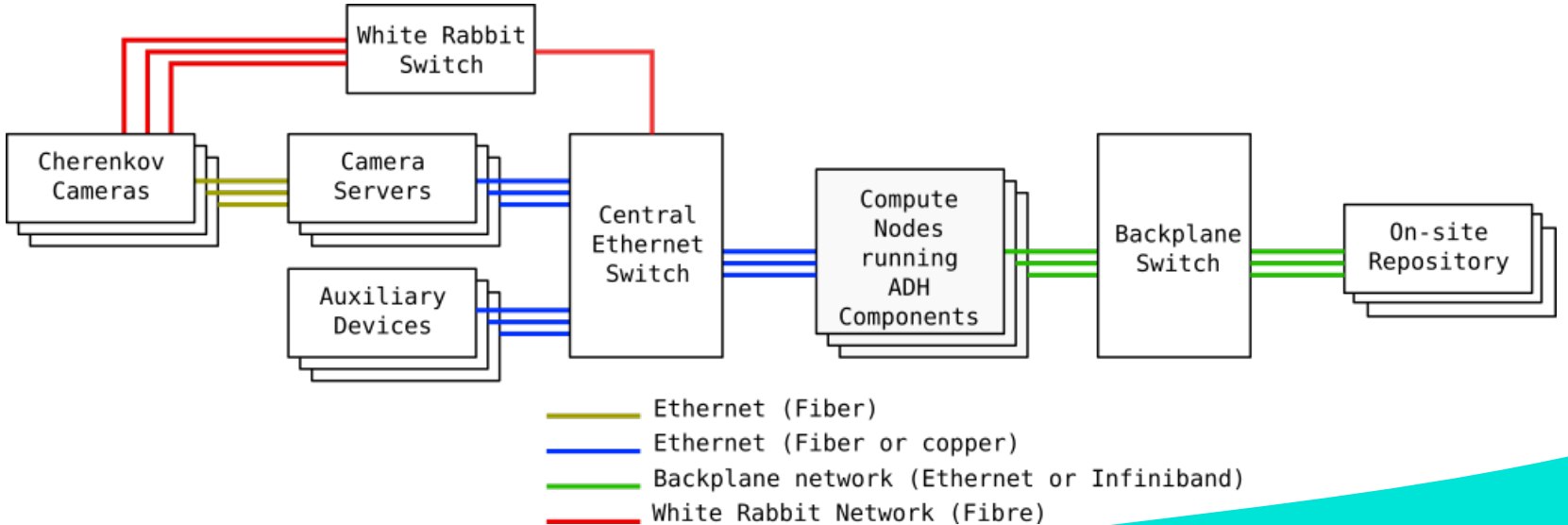
- ACADA Governs all non-safety-critical automatic on-site operations.
- Scheduling Blocks (SBs) are submitted by the Short-term Scheduler (STS), resources are allocated by the Resource Manager (RM), and commands are issued by the Central Control (CC).
- 99 telescopes in addition to numerous additional instruments.
- Reaction times on the few seconds timescales.
- Manage the simultaneously operating or up to 8 subarrays, with some shared resources (e.g. shared LIDAR jumping from one subarray to another).
- Implements different operation strategies (e.g., wobble mode, on-off, scans).



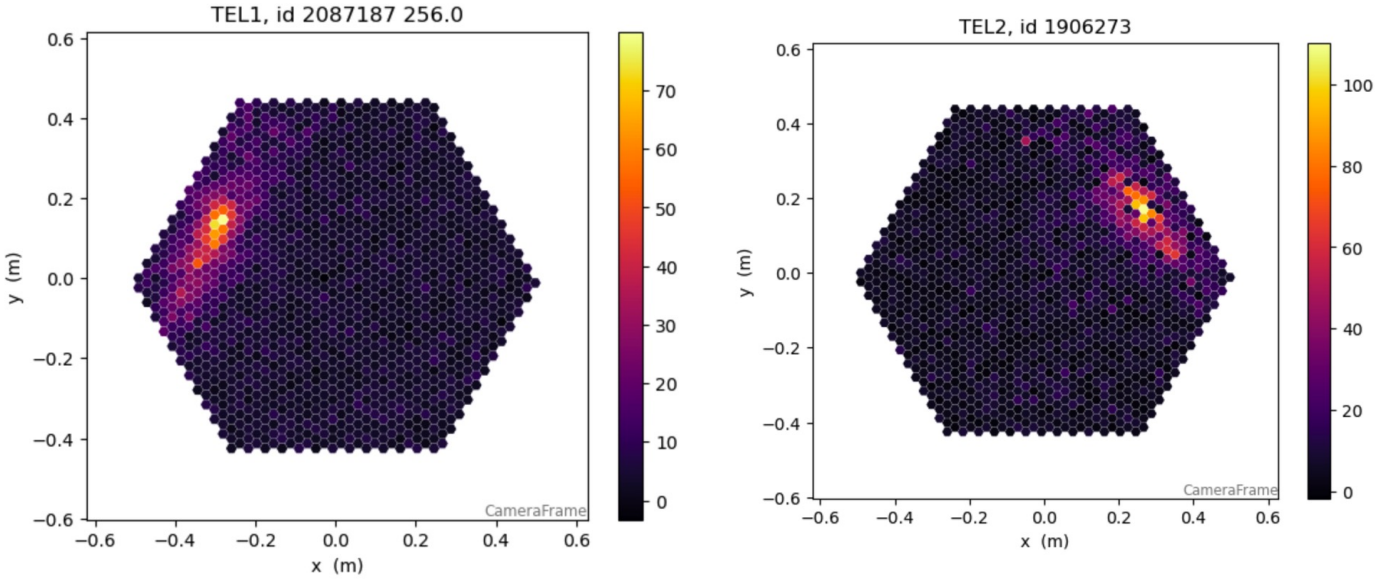
Data Handling

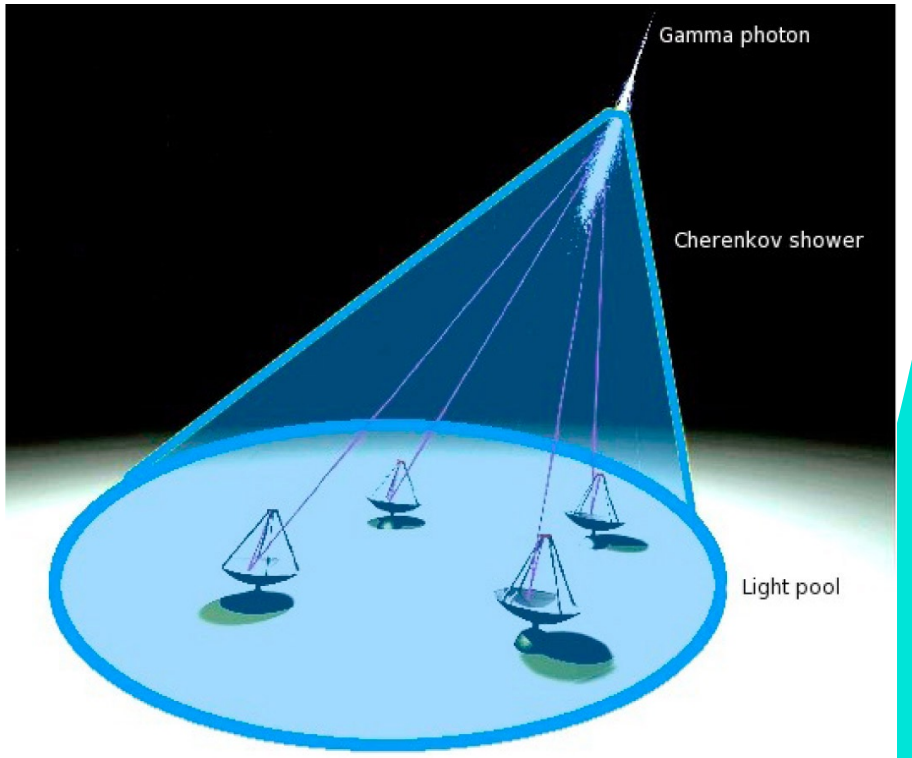
- The Array Data Handler (ADH) is responsible for receiving and storing scientific data from the telescopes.
- Receive trigger timestamps from the cameras and send back confirmation from the central trigger (after timestamp comparison).
- Reduce online the volume of received data.
- Hand over the stored data to the Data Processing and Preservation System (DPPS).

Data throughput	24 Gb/s/LST (up to four)
	12 Gb/s/MST (up to 25)
	2 Gb/s/SST (up to 70)
Individual telescope trigger rates	15 kHz/LST (up to four)
	14 kHz/MST (up to 25)
	1.2 kHz/SST (up to 70)
Array-level trigger rates	40 kHz
Dead time after a coincidence	of 250 ns
DVR factor	Up to 50



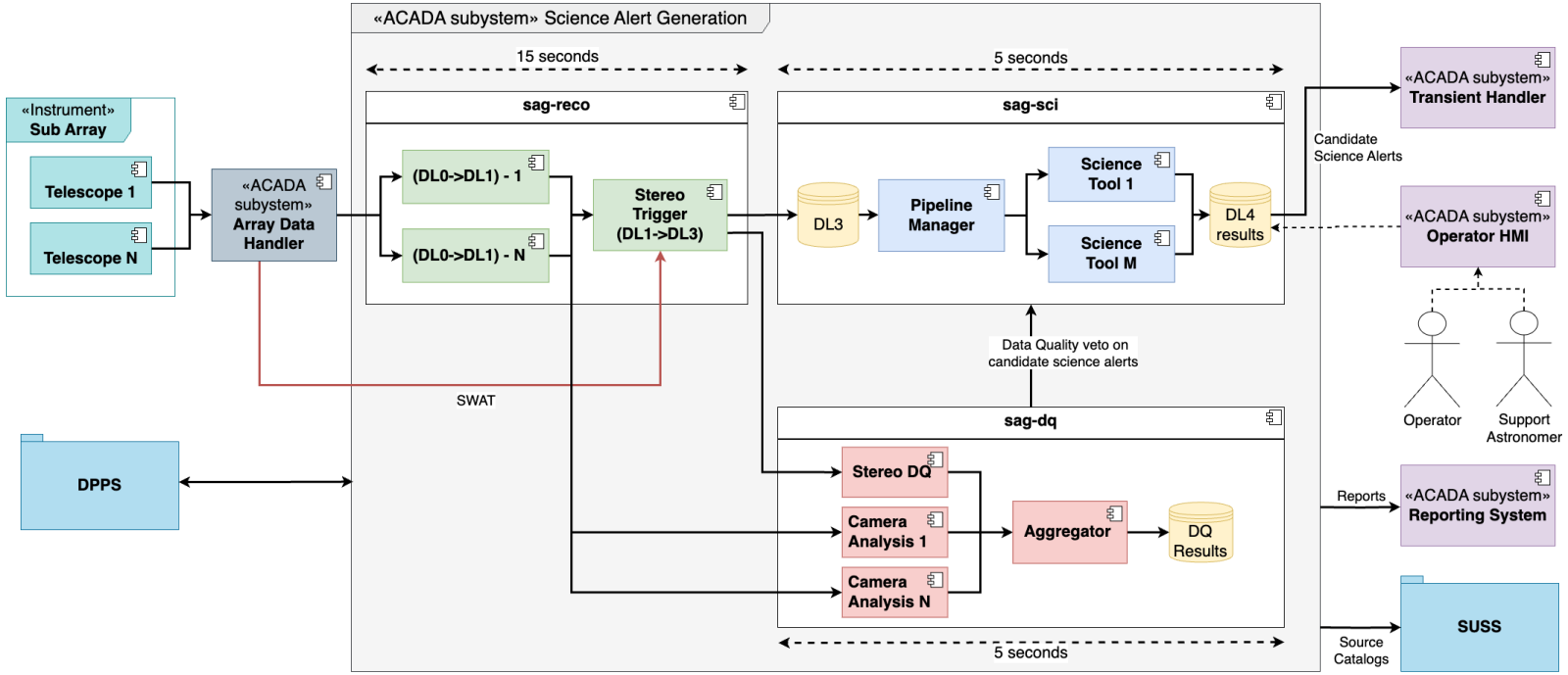
SWAT: Software Array Trigger

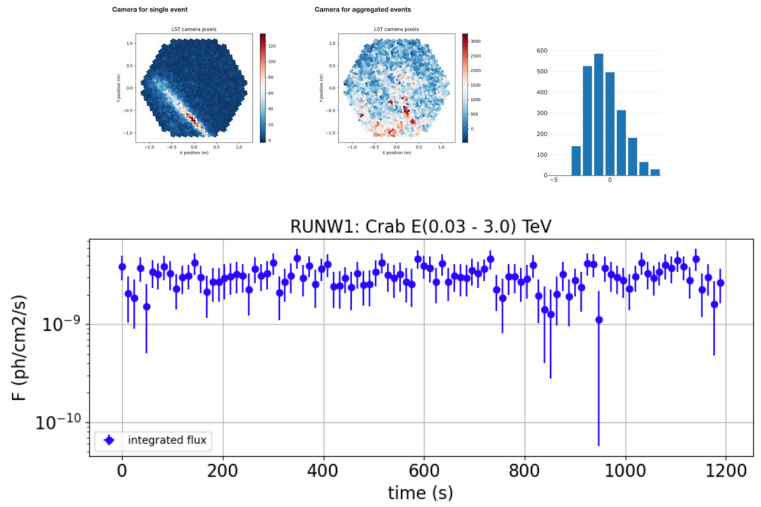


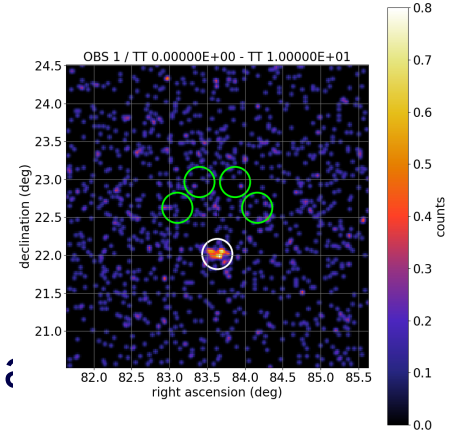


- SWAT: The software component used to match telescope-level triggers into sub-array triggers by examining hardware-generated high-precision timestamps.

ACADA – SAG Pipeline

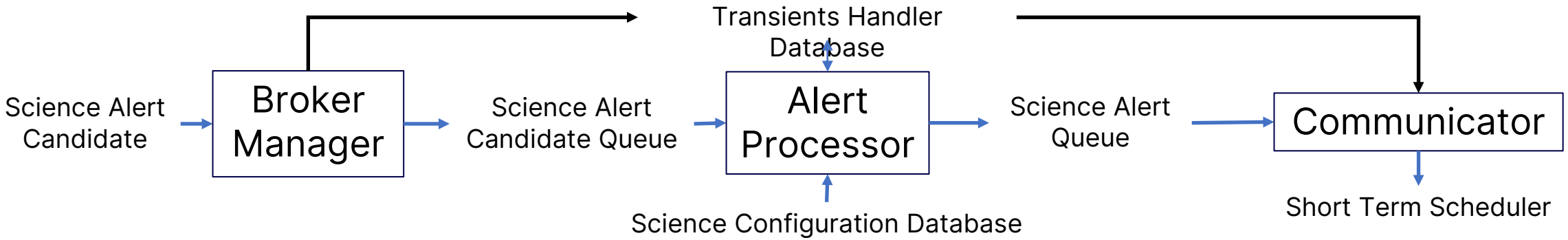






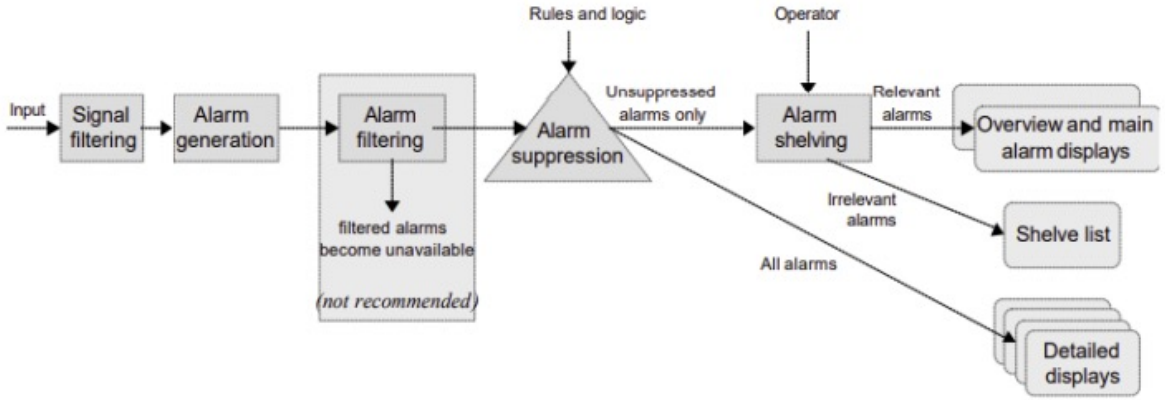
- Analysis pipeline running online (*Category A pipeline*).
- real-time data quality check.
- Produce quick-look and data quality summaries, and data qua
- Produce internal Science Alert candidates

The Transients Handler



- The Transients Handler (TH) is the subsystem of ACADA that will provide the means of handling alerts and schedule follow-up observations.
- External, internal alerts.
- Prompt or delayed alerts.
- Not every alert has priority overt he ongoing observations.

The Array Alarm System



Ref.: Norwegian Petroleum Directorate

1. *Signal filtering*: processing raw input signals to remove signal noise and other information that is not important, such as small, rapid oscillations.
2. *Alarm generation*: comparing the input signal with signal limits and checking the process and system states.
3. *Alarm filtering*: preventing an alarm signal reaching the operator (Disable an alarm).
4. *Alarm suppression*: preventing an alarm from being presented in main alarm displays, e.g. overview displays, but the alarm is still available in the system at a more detailed level
5. *Alarm shelving:* manually removing an alarm from the main list and placing it on a shelve list, temporarily preventing the alarm from re-occurring on the main list until it is removed from the shelf.

- Alarms have an associated ”database”, indicating the expected to procedure by the operators.
  - Some alarms may only require acknowledgement, so others my trigger very intense activity.
- Every time an alarm is triggered will require attention of the operators → Need proper configuration.
- Efficient addressing of alarms is critical for the observatory KPIs.

Conclusions

- We have presented CTAO's array operations from different perspectives
  - The CTAO arrays day and night operations, and the crew that uses it.
  - The CTAO system composition as arrays of Cherenkov Telescopes.
  - Using the CTAO Arrays for acquiring science data.
  - Array Operations – the Software perspective - How to design and implement the necessary software?