ALMA Observatory

The ALMA Software and Release Management

Ruben Soto
Software Operations Group & Release Manager
Joint ALMA Observatory

ALMA Observatory

Agenda

1. ALMA Software Components (Structure and Division)
2. Software Delivery Process (History, Present and Evolution)

ALMA Observatory

1. ALMA Software Components

ALMA Software Components

- ALMA software components cover end to end system to operate the observatory.
  - From the cradle…
    - Proposal Preparation
    - Proposal Review
  - through infancy to adulthood …
    - Program Preparation
    - Dynamic Scheduling of Programs
    - Observation
    - Calibration & Imaging
    - Data Delivery & Archiving
  - up to afterlife:
    - Archival Research & VO Compliance

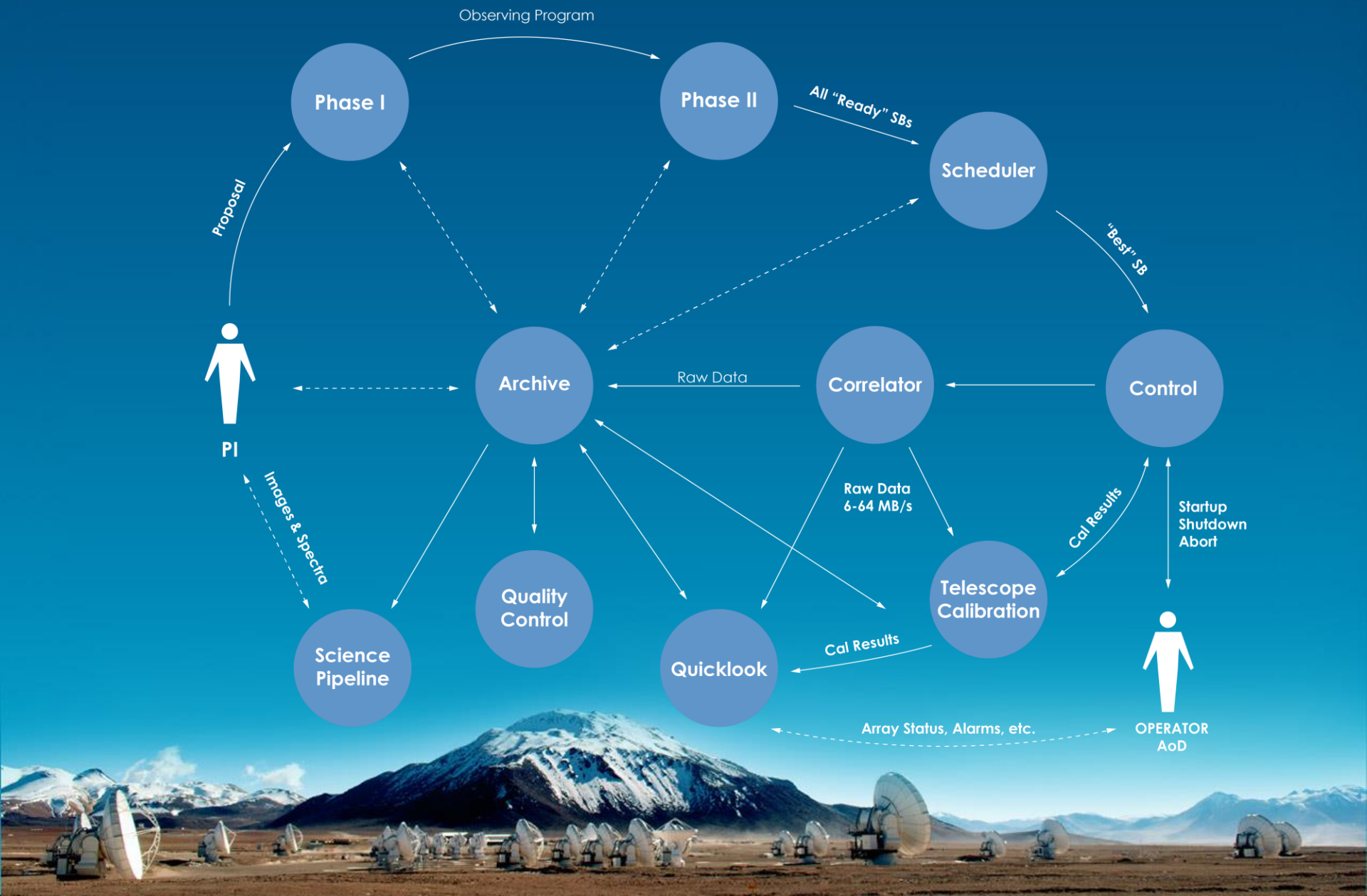
ALMA Software Components

- ALMA software components are classified as part of ONLINE or OFFLINE subsystems by depending of their functionality.

ALMA Observatory

Data flow

NINS



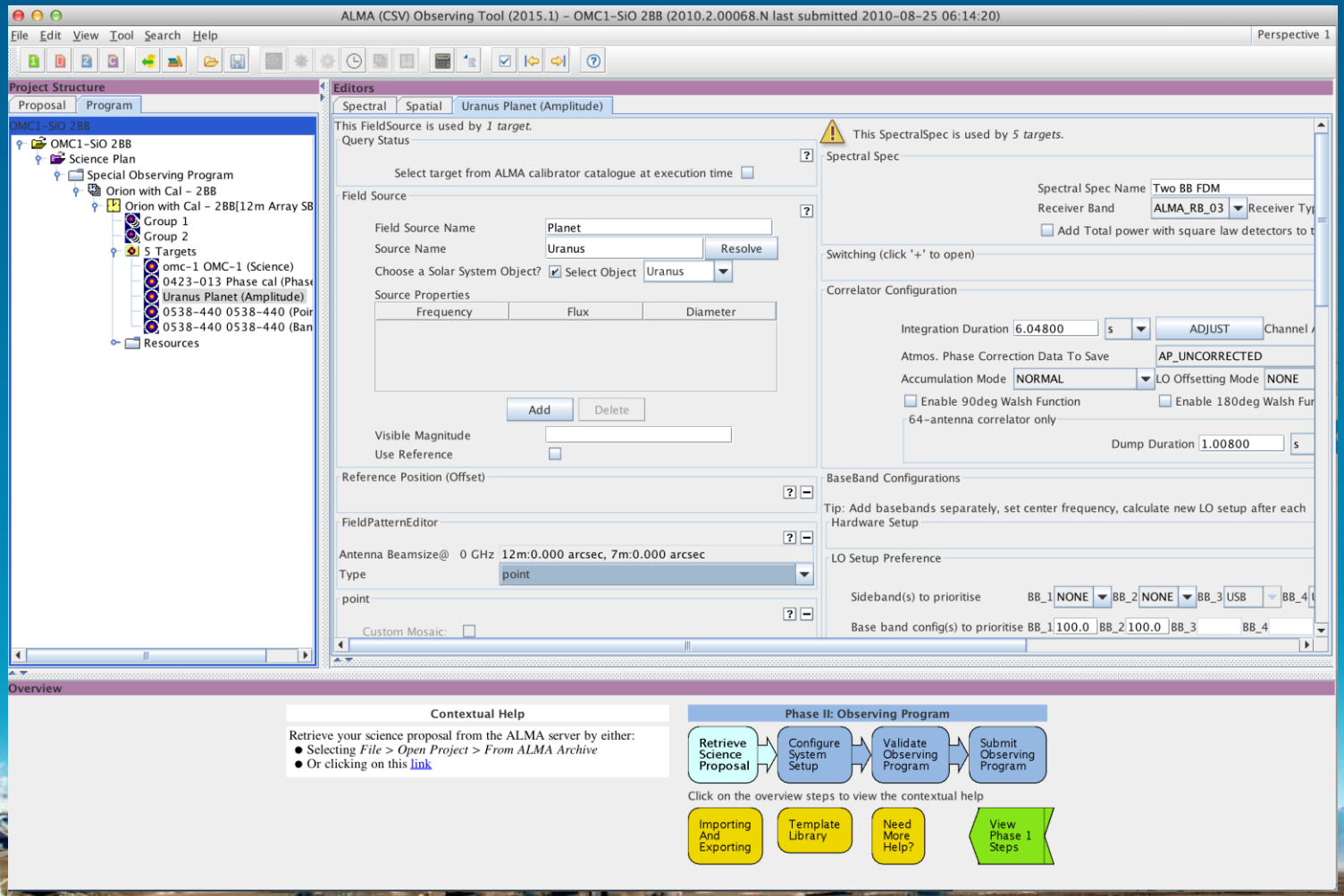
ALMA Observatory

Offline Software

- Proposal submission, project rating and preparation, projects lifecycle and general observatory interfaces are considered part of the OFFLINE software.
- Several web tools have been created for different purposes. They are basically:
  - Based in Java Servlet technology (and ZK platform).
  - Using Apache tomcat as servlet container.
  - Using https protocol for data security.
- Some examples of offline tools are..

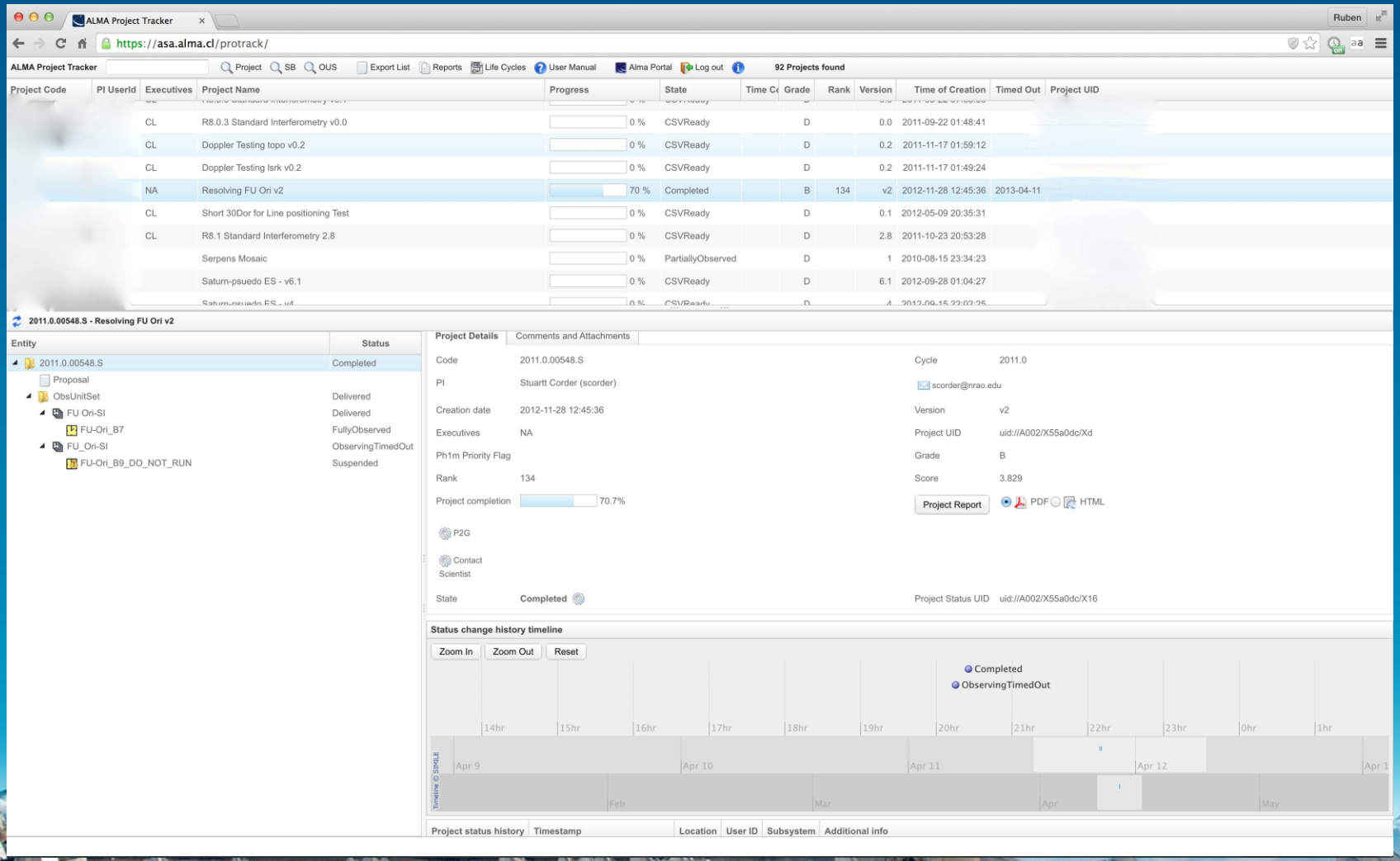
Offline Tools

- **ALMA-OT**: Proposal submission and observation preparation



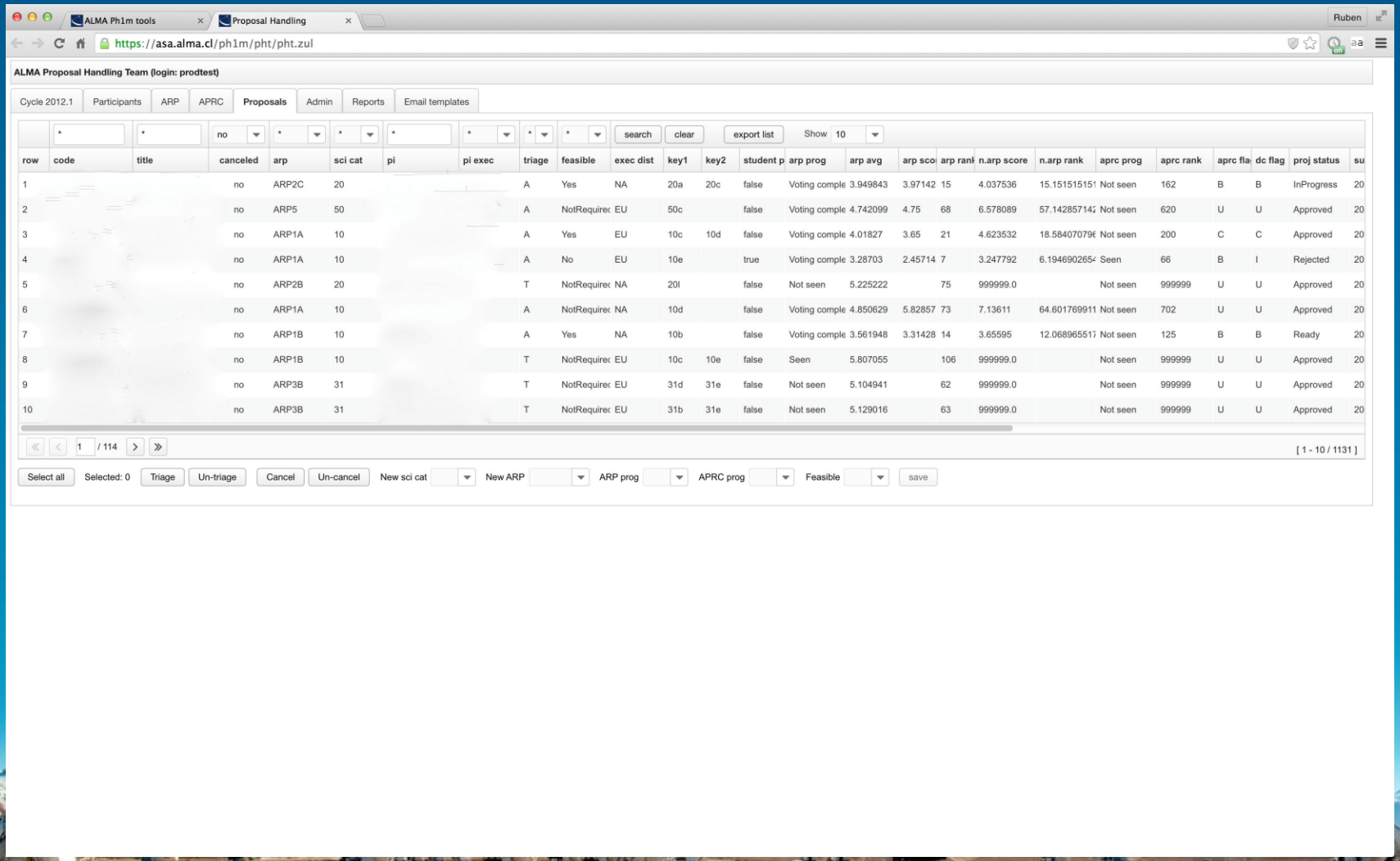
Offline Tools

- **Project Tracker**: Tracking project lifecycle



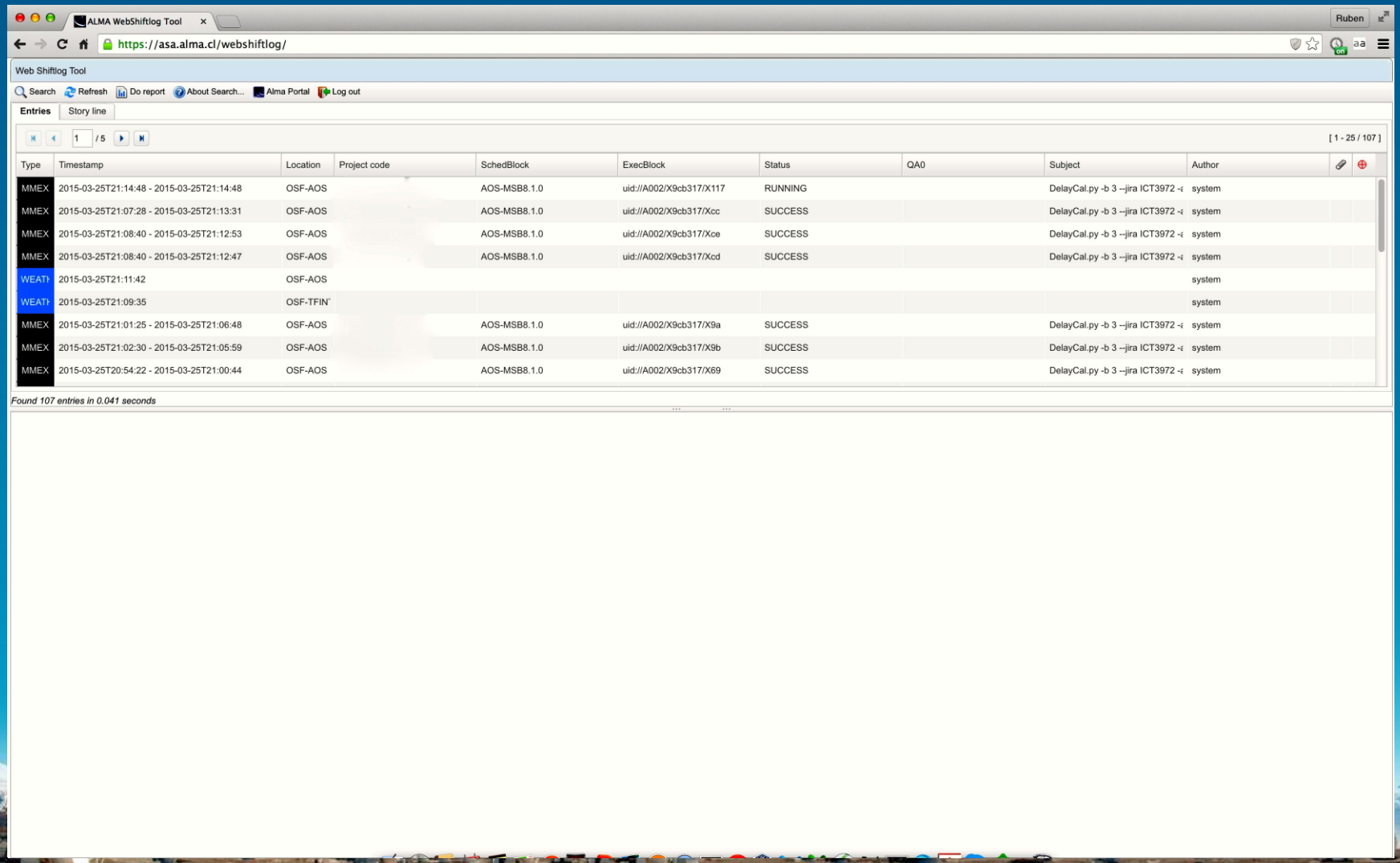
Offline Tools

- **Phase 1 Manager (ph1m)**: Project rating



Offline Tools

- **Shift Log (Web)**: Tracking observatory operations



ALMA Observatory

Online Software

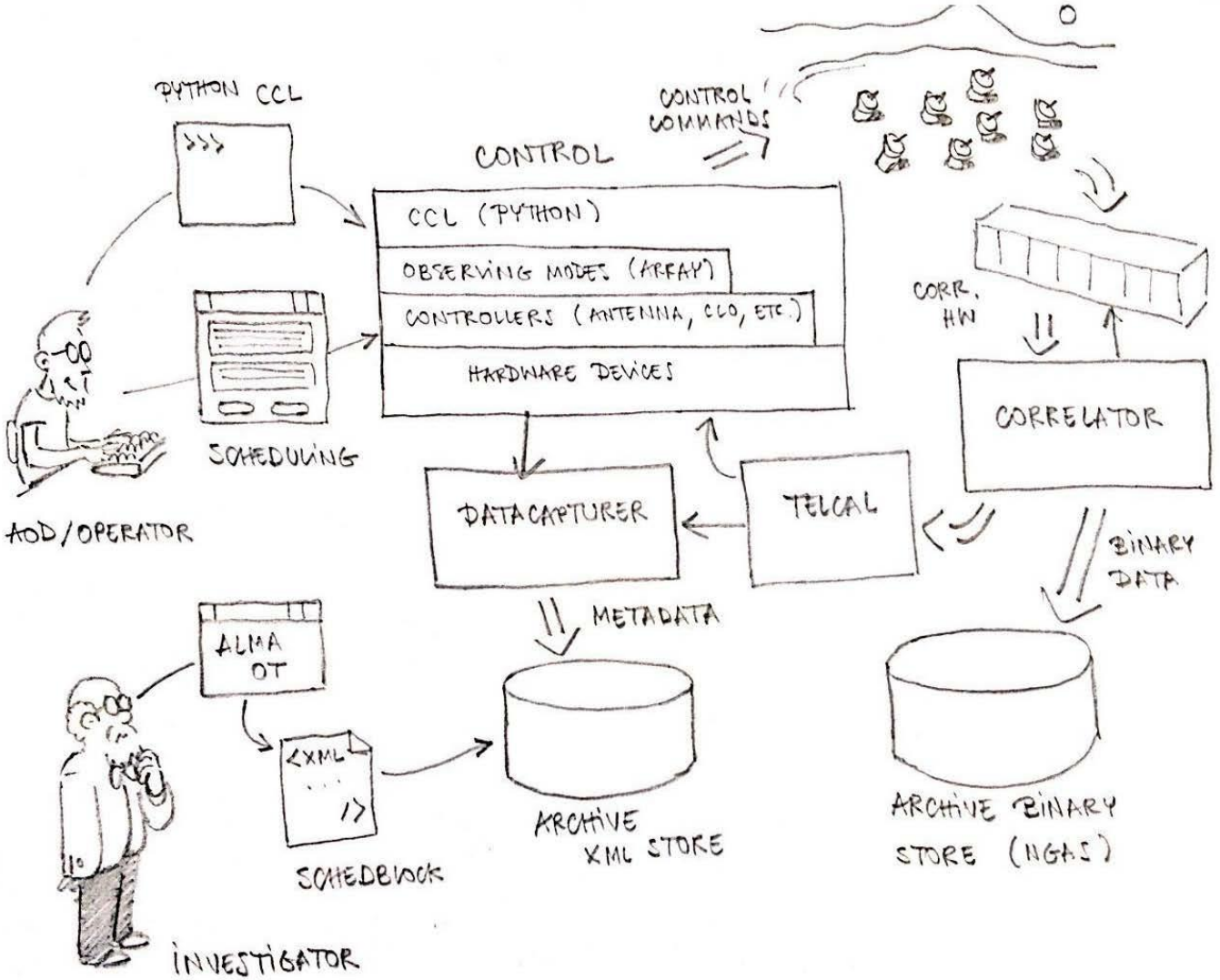
- The purpose of the online software is to monitor and control the ALMA telescope and to execute scheduling blocks from approved scientific observing projects.
- Online software (applications) are implemented:
  - Over distributed framework based on CORBA and developed for ALMA Observatory: ALMA Common Software (ACS).
  - ACS provides component communication, error and exception handling, alarm system, etc.
  - Applications are programmed on C++, Java and Python languages
  - Hardware response is managed by real time operating system (RTAI).

Online Software

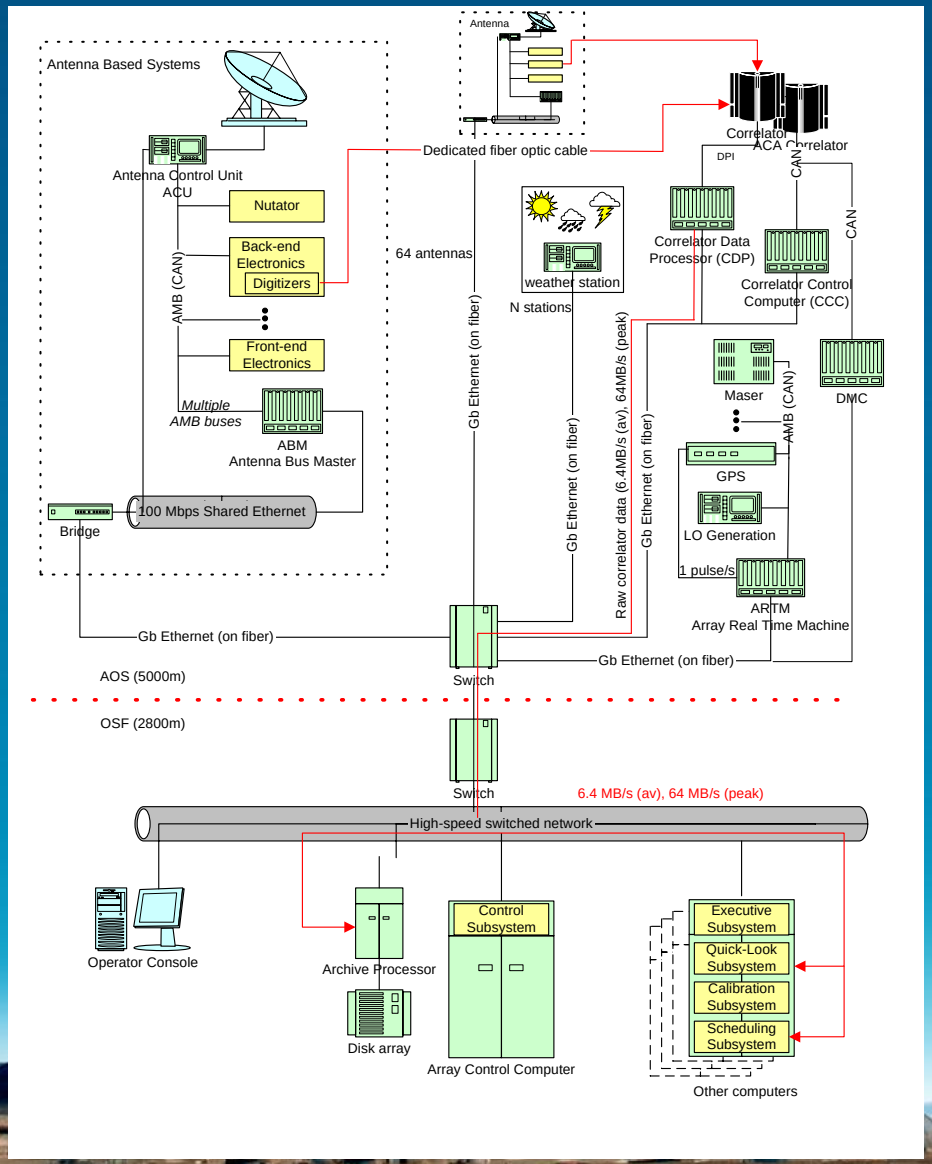
- Online software consider the following subsystems: ACS, CONTROL, BL CORR, ACA CORR, TELCAL, SCHEDULING and PIPELINE Infrastructure.

- PIPELINE heuristics and data reduction software (CASA) are not considered as part of ALMA Software

… In More Detail



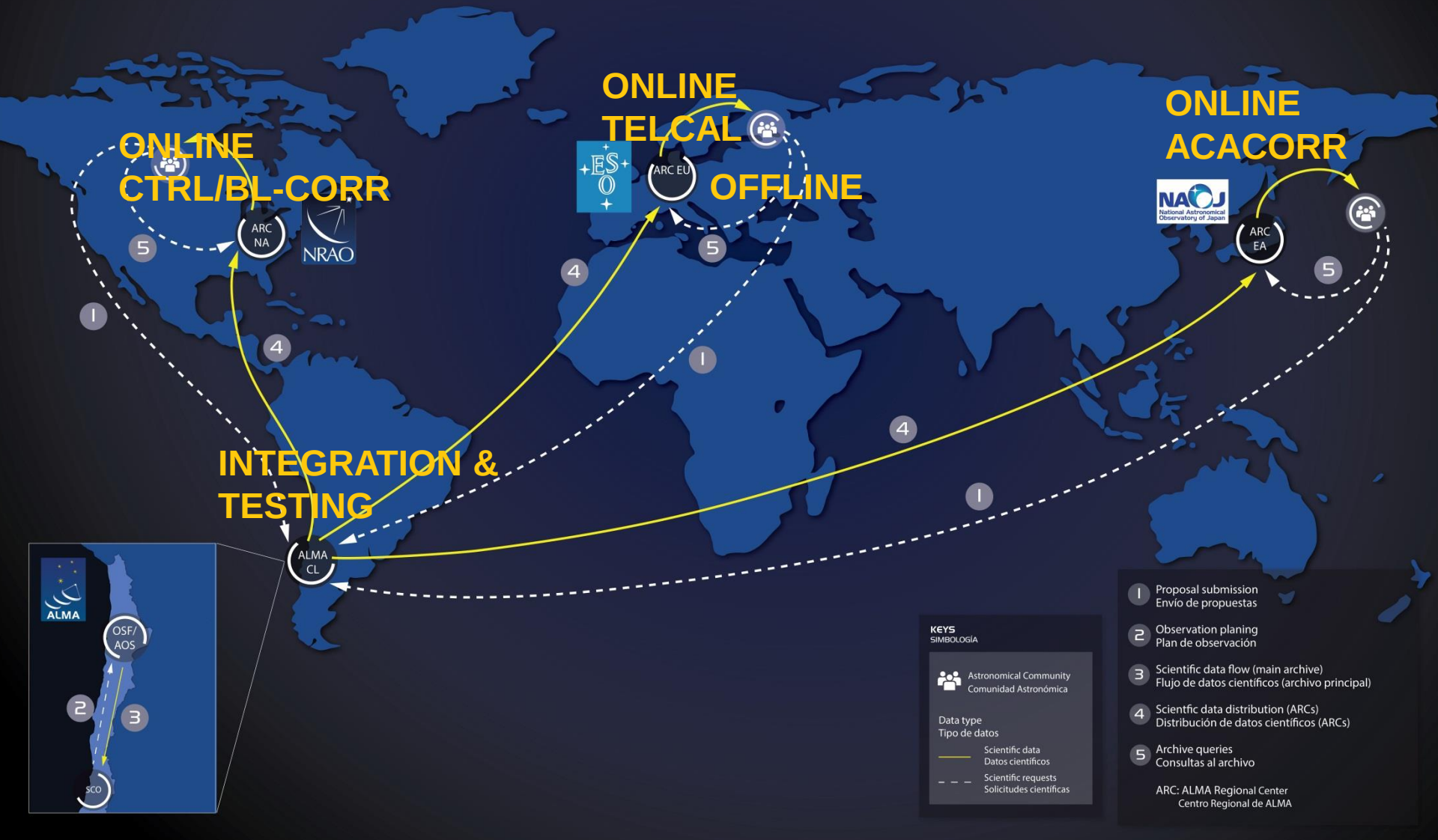
Online Software Architecture (physical)



ALMA Observatory Inauguration

ALMA Worldwide Data Flow
Flujo de Datos Mundial de ALMA

The ALMA Development and Operations Centers



ALMA Observatory

2. Software Delivery Process

Beginning of ALMA

- Software releases were delivered every 6 months, each one contained lots of new functionality at ONLINE and OFFLINE sides.
- Testing was executed at the developer centers and integration was done at the operational site.
- Testing was focused only at the core functionality (regression tests) but not at the new one added. New functionality verification was delegated to science groups.

Beginning of ALMA

- Simulation was not mature to isolate a significant number of integration problems! -> Several errors were found when using production hardware (i.e. array elements!) increasing the cost of correcting them.
- Too many changes were introduced while integrating, adding instability to a already fragile system.
- Release stabilization took months before being ready for science commissioning.

Introducing Changes

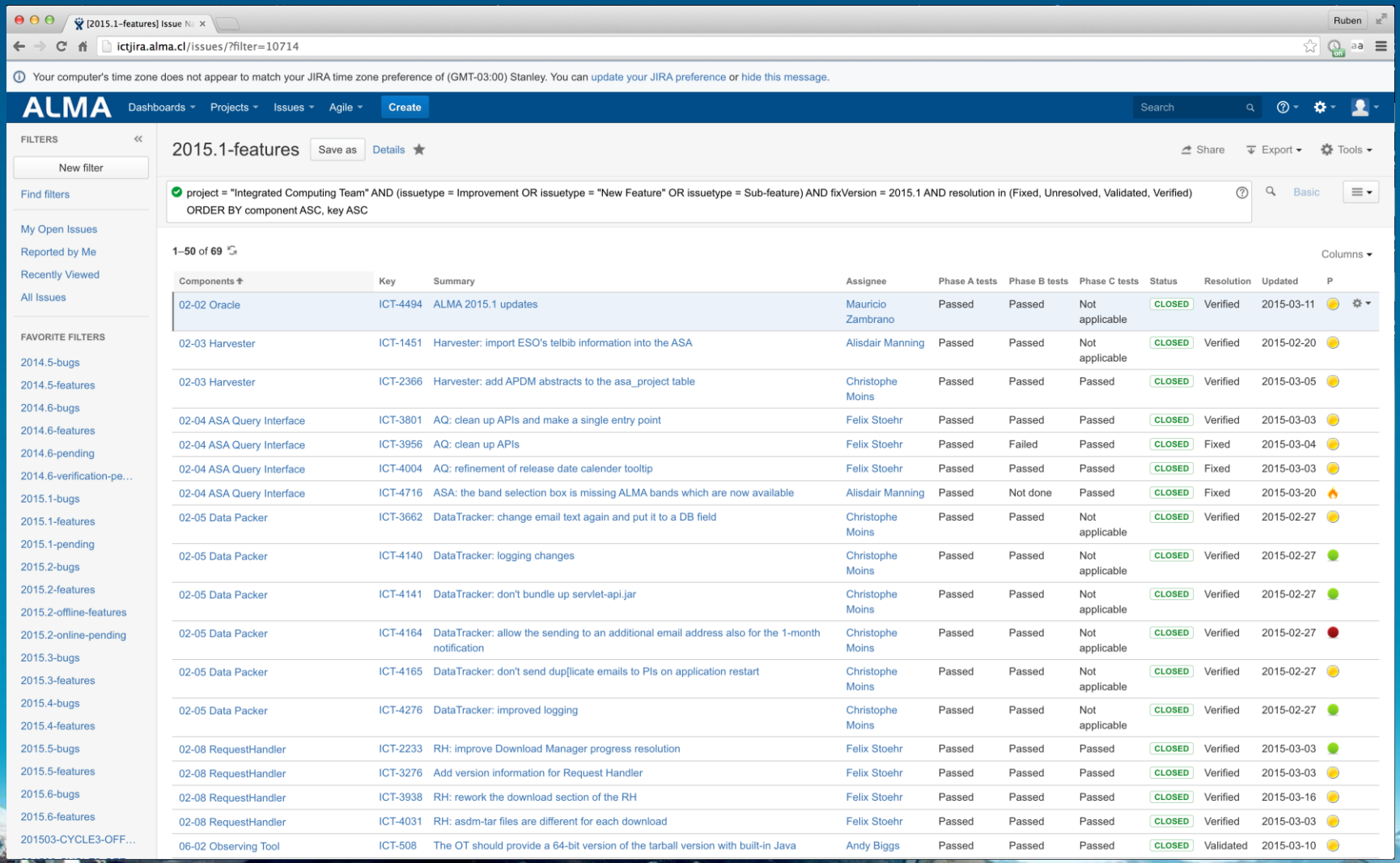
- Increase delivery frequency
  - Originally from every six months to bi-monthly schema (including testing and integration)
- Reduce scope per release
  - Features must be ordered according to observatory milestones (proposal submission, project rating, start cycle observations, etc).
- Define clear software delivery phases adding a formal handover between each phase
  - Introduce two (new) roles: Release Manager and Acceptance Manager

Introducing Changes

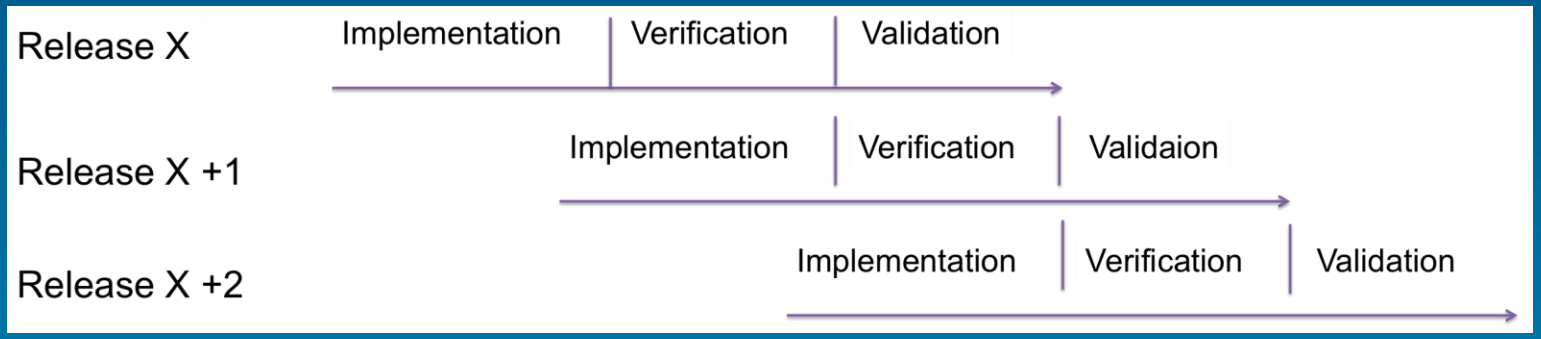
- Define clear software delivery phases adding a formal handover between each phase
  - Phase A: Implementations and developer tests
  - Phase B: Verification by Integration & Testing Team (Computing)
  - Phase C: Validation by commissioning team (Science)
- Formalize negotiation about release contents, deadline and software acceptances
  - Introduce two roles: Release Manager (Computing) and Acceptance Manager (Science)

ALMA Observatory

Example of Release Content



Incremental Release Process (Today)



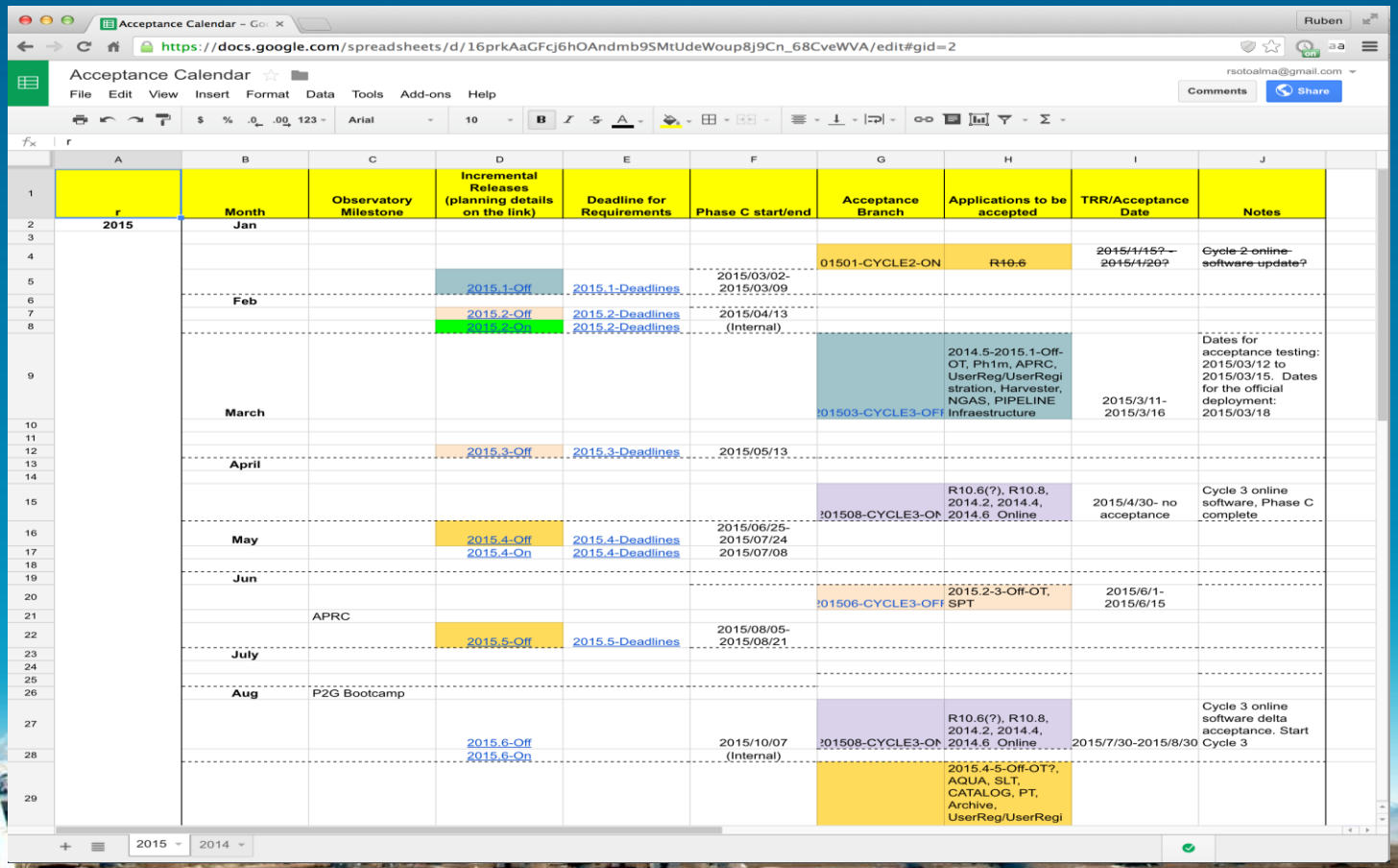
- Incremental process allowed also optimizing tests resources, including developers, testers, and client testers
- Release Manager is responsible of incremental release calendar and negotiate release contents with acceptance manager

Software Acceptances

- Incremental releases are accepted for science milestones (cycle-X proposal, project rating, start cycle-X observing) :
  - According to observatory's milestones
  - Incremental releases have passed verification and validation phases
  - Acceptances tests have been successfully carried on.
- Acceptance Process has the following steps:
  - Test Report Review
  - Acceptance Testing
  - Acceptance Meeting
  - Software deployment on production environment (JAO and ARCs)

Software Acceptances

- Acceptance Manager is responsible for creating acceptance calendar according to observatory milestones.



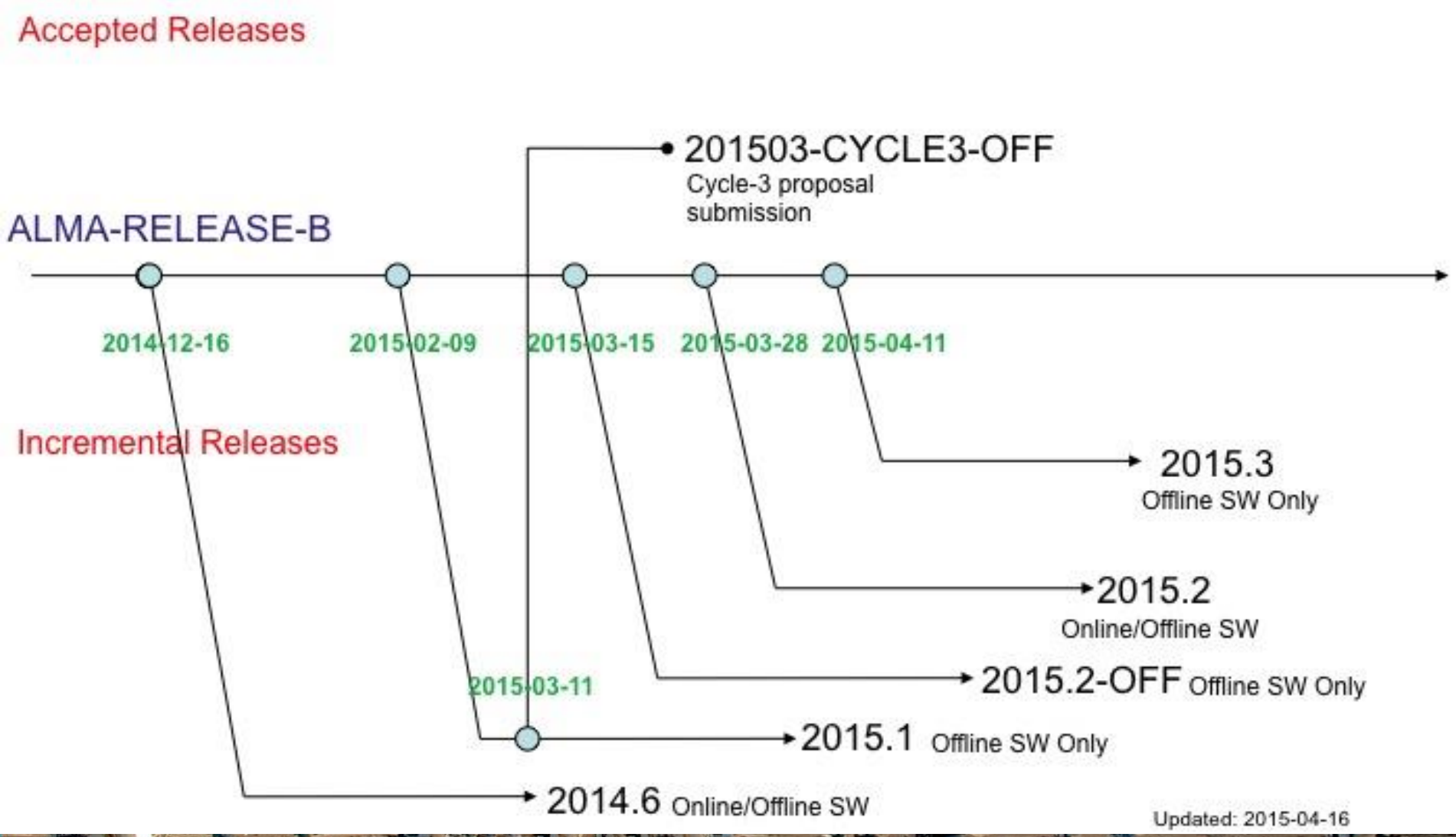
Acceptance Calendar

https://docs.google.com/spreadsheets/d/16prkAaGFcj6hOAndmb9SMtUdeWoup8j9Cn_68CveWVA/edit#gid=2

r	Month	Observatory Milestone	Incremental Releases (planning details on the link)	Deadline for Requirements	Phase C start/end	Acceptance Branch	Applications to be accepted	TRR/Acceptance Date	Notes
2015	Jan								
						01501-CYCLE2-ON	~~R10.6~~	~~2015/1/15? - 2015/1/20?~~	~~Cycle 2 online software update?~~
			2015.1-Off	2015.1-Deadlines	2015/03/02-2015/03/09				
	Feb								
			2015.2-Off	2015.2-Deadlines	2015/04/13 (Internal)				
			2015.2-On	2015.2-Deadlines					
							2014.5-2015.1-Off-OT, Ph1m, APRC, UserReg/UserRegistration, Harvester, NGAS, PIPELINE Infraestructure		Dates for acceptance testing: 2015/03/12 to 2015/03/15. Dates for the official deployment: 2015/03/18
	March					01503-CYCLE3-OFF		2015/3/11-2015/3/16	
			2015.3-Off	2015.3-Deadlines	2015/05/13				
	April								
						01508-CYCLE3-ON	R10.6(?), R10.8, 2014.2, 2014.4, 2014.6 Online	2015/4/30- no acceptance	Cycle 3 online software, Phase C complete
			2015.4-Off	2015.4-Deadlines	2015/06/25-2015/07/24				
	May		2015.4-On	2015.4-Deadlines	2015/07/08				
	Jun								
						01506-CYCLE3-OFF	2015.2-3-Off-OT, SPT	2015/6/1-2015/6/15	
		APRC							
			2015.5-Off	2015.5-Deadlines	2015/08/05-2015/08/21				
	July								
	Aug	P2G Bootcamp							
						01508-CYCLE3-ON	R10.6(?), R10.8, 2014.2, 2014.4, 2014.6 Online	2015/7/30-2015/8/30	Cycle 3 online software delta acceptance. Start Cycle 3
			2015.6-Off		2015/10/07 (Internal)				
			2015.6-On						
							2015.4-5-Off-OT?, AQUA, SLT, CATALOG, PT, Archive, UserReg/UserRegi		

2015 2014

Incremental and Accepted Release Schedule



Software Requests

- Once a release is accepted for science observations all changes are controlled.
  - Software Configuration Control Board (SCCB) is responsible for approving or rejecting software requests.
  - SCCB is compound by science, engineering and computing representatives
  - SCCB meets once per week for resolving software requests.
- Software requests involves:
  - Database persistent model changes.
  - Software patches
  - Service releases (whole application updates)
  - Change requests (changes at the software tools)

The Future

- ALMA is transitioning from construction to operations:
  - Less access (technical time) to the operational HW (testing).
  - Continuous operation will require software robustness instead of new functionality.
  - Software should be more stable.
- Operational model will impact SW delivery process:
  - Less testing time with access to operational HW.
  - Simulation capabilities should be improved.
  - Number of features will decrease per release.
  - A more agile paradigm can be implemented.

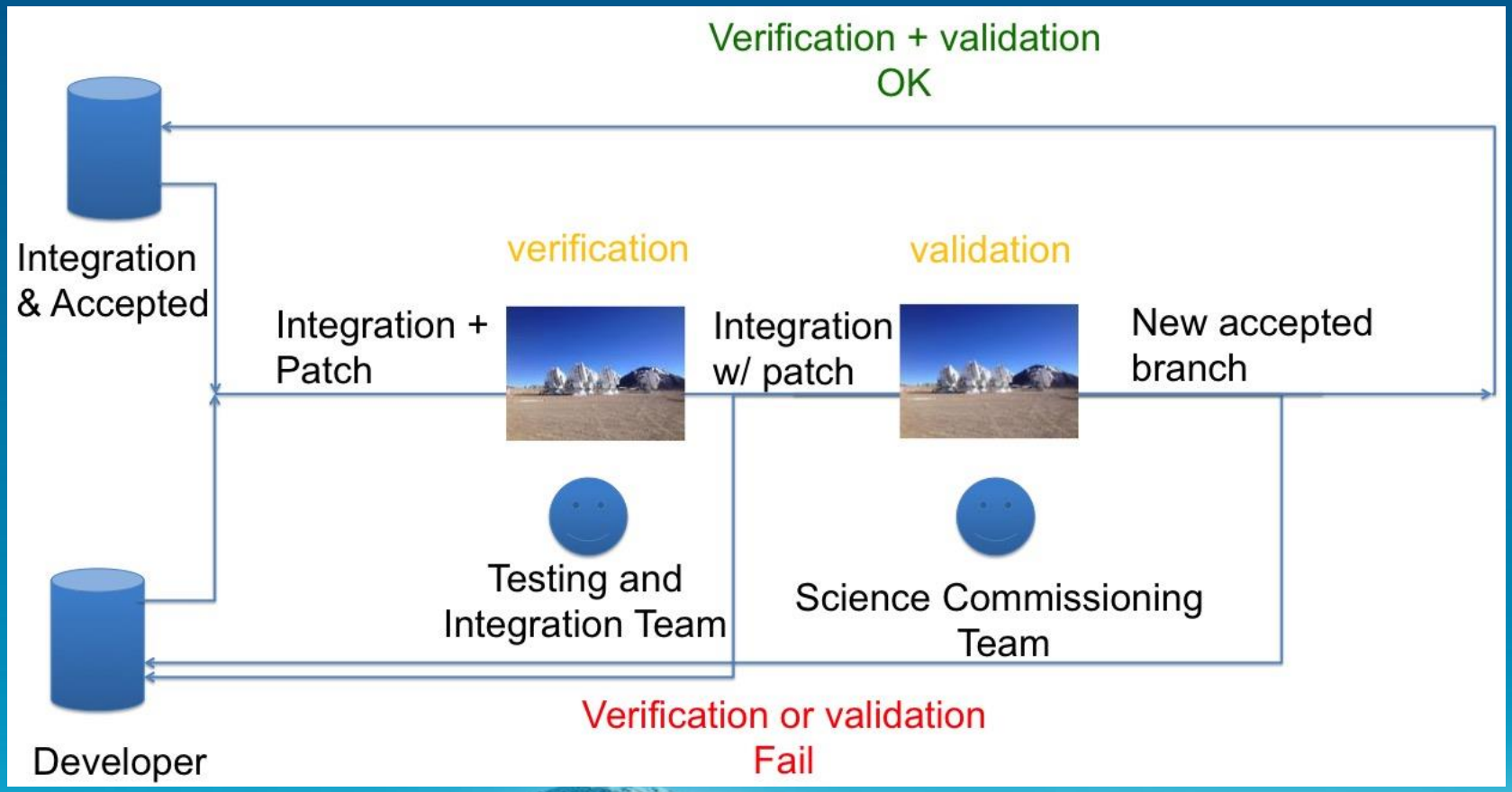
Transition to Agile Approach

- Continuous integration model:
  - There is a stable (“accepted”) branch used for science observations.
  - Developers commit new features in their local repository (or branches) as a software patch compatible with integration branch.
  - Patch is scheduled to be verified by Integration & Testing team during a time slot of a technical time.
  - If feature passed verification; it is committed into the integration branch. Otherwise, patch is rejected to be integrated and scheduled for a new technical time.

Transition to Agile Approach

- Continuous integration model:
  - After verification, feature must be validated by Science group from integration branch.
  - If validation success, a new accepted branch is created from the integration branch and used for science observations.
  - Otherwise, accepted branch remains the same until fix problems with the new features.
  - Note this model does not require acceptance process but…

Agile Approach Workflow



Considerations

- This model will require more discipline from developers, testers and science commissioning team.
  - Developers must schedule the features according to Observatory milestone well in advance (specially big or difficult ones)
  - Verification phase start early in order to get the features working fine for validation.
  - High (and quick) interaction between tester and developers is expected
  - Scientists must be available for complete validation and get the new software ready for science observations.

ALMA Observatory

Considerations

- A verification phase under simulated environment can be added preliminary to the operational HW testing (online system).
- Online system validation usually takes many time before be declared as success, so it will require special validation period (less “agile”) .
- Offline software still produced big number of features improvements.

Thanks! Questions?

Further information:
http://www.almaobservatory.org

Contact point: rsoto@alma.cl

The Atacama Large Millimeter/submillimeter Array (ALMA), an international astronomy facility, is a partnership of Europe, North America and East Asia in cooperation with the Republic of Chile. ALMA is funded in Europe by the European Organization for Astronomical Research in the Southern Hemisphere (ESO), in North America by the U.S. National Science Foundation (NSF) in cooperation with the National Research Council of Canada (NRC) and the National Science Council of Taiwan (NSC) and in East Asia by the National Institutes of Natural Sciences (NINS) of Japan in cooperation with the Academia Sinica (AS) in Taiwan. ALMA construction and operations are led on behalf of Europe by ESO, on behalf of North America by the National Radio Astronomy Observatory (NRAO), which is managed by Associated Universities, Inc. (AUI) and on behalf of East Asia by the National Astronomical Observatory of Japan (NAOJ). The Joint ALMA Observatory (JAO) provides the unified leadership and management of the construction, commissioning and operation of ALMA.