ALMA Science Operations and User Support (Software)

Mark G. Rawlings[a], Lars-Ake Nyman[a] and Baltasar Vila Vilaro[a]

[a] The Joint ALMA Observatory, Alonso de Cordova 3107, Vitacura, Santiago, Chile

ABSTRACT

An overview will be presented of the various software subsystems currently in development for the support of ALMA Early and Full Science Operations. This will include a description of the software subsystems currently being devised to address the following: Proposal preparation and submission system (ObsPrep); Software systems for tracking the proposal review process, post-acceptance project tracking, plus other miscellaneous components (ObOps); Observation Scheduling (Scheduler); Data Archive (Archive); Data Reduction Pipelines (QuickLook, Pipeline); Quality Assurance and Trend Analysis (AQUA). Additional user support systems (Science Operations Web Pages, User Portal, etc.) will also be outlined.

Keywords: ALMA, software, proposals, telescope operations

1. INTRODUCTION

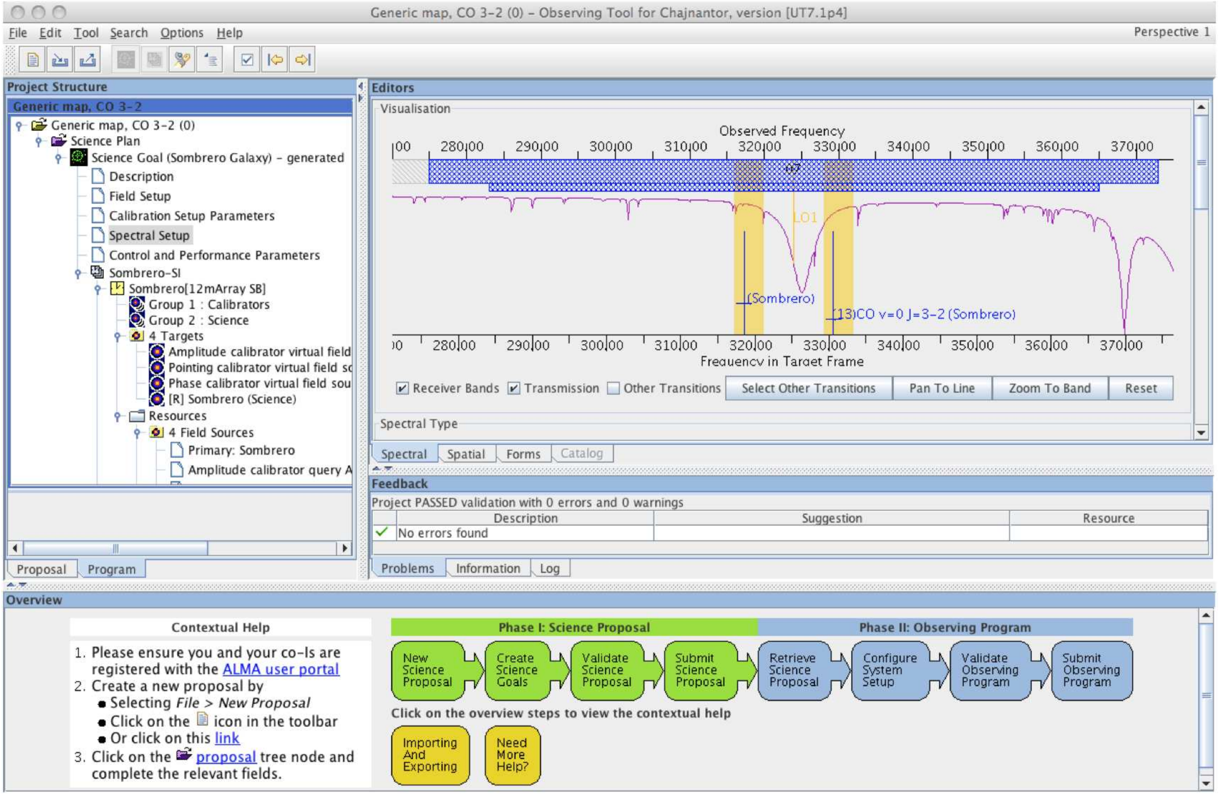
In order to run any modern large observatory, including the Atacama Large Millimeter/submillimeter Array (ALMA), a large amount of software is needed. For ALMA, in addition to the multilevel real-time control software that controls the actual hardware such as the antennas, receivers, correlators, etc., a significant amount of other software is needed in order to differentiate between a working telescope and a working observatory. Here, we present an overview of the various software subsystems that are being developed in order to accomplish this. A more detailed description of these systems is given in the ALMA DSO Implementation Plan document.[1] Using the software components presented herein, the ALMA observatory will be able to support a large-scale proposal review process, automatically schedule the execution of the successful projects, reduce the acquired data, track the completion status of each project, and deliver the necessary final data products to the Principal Investigators (PIs) of each such project. Mirroring the structure of the ALMA project as a whole, these software subsystems must all interact with each other seamlessly, despite being developed (and in some cases operated) in locations scattered around the world. The offline data reduction package Common Astronomy Software Applications (CASA), will not be discussed in detail in this article, as it has already (justifiably) been made the subject of numerous articles and meetings in its own right.

2. OBSPREP

Development Lead: Alan Bridger (ATC, Edinburgh)

The Observation Preparation subsystem (ObsPrep) development team provides the software needed for the preparation and submission of ALMA proposals The basic software architecture involved is a client/server arrangement. The client is a user-downloadable program called the ALMA Observing Tool (commonly referred to as the AOT or OT), and the server is a program running on a Santiago-based Linux rack server system. The details of these two software components are discussed below.

Further author information: (Send correspondence to M.G.R.)
M.G.R.: E-mail: mrawling@alma.cl, Telephone: +56 2 467 6261



Project PASSED validation with 0 errors and 0 warnings

	Description	Suggestion	Resource
✓	No errors found		

Contextual Help

1. Please ensure you and your co-Is are registered with the ALMA user portal
2. Create a new proposal by
   - Selecting *File > New Proposal*
   - Click on the 📄 icon in the toolbar
   - Or click on this link
3. Click on the 📂 proposal tree node and complete the relevant fields.

Figure 1. A screenshot of the ALMA Observing Tool. The program tree structure is displayed in the upper-left pane, the spectral editor window is shown in the upper-right "Editors" pane and the feedback pane is displayed directly below the "Editors" pane. The clickable contextual help interface is shown in the bottom pane.

2.1 ALMA Observing Tool (AOT/OT)

The ALMA OT is a client program based on Java that is downloaded by all ALMA users via an automated Java Web Start deployment (other deployment options are also available). The Java platform was chosen as it offers a "write once, run anywhere" approach, allowing a single client program to be deployed to a user base running a heterogeneous range of operating systems (e.g. GNU/Linux, Mac OS X and MS Windows). The Java Web Start technology allows the download and installation process to be simplified to the point of just requiring the user to follow a single download link on a web page, and also permits the automated rollout of upgrades. The use of a locally-stored client program allows ALMA observers to prepare proposals without requiring a full-time connection to the internet.

The ALMA OT is based around a WIMPs (Windows, Icons, Menus and Pointers) GUI*. The aim is to provide a user-friendly way of specifying all the technical details required for any given ALMA science proposal. The normal user interface, an example of which is shown in Fig. 1, primarily consists of a single large window containing a number of discrete panes, as follows:

- A pane displaying a schematic tree diagram of the overall hierarchical structure of the science proposal.
- A pane displaying various editor interfaces, the content of which depends on the item selected from the tree diagram. These include a number of electronic forms, the fields of which together specify all the parameters of the proposal, plus two interactive GUIs (spatial and spectral) that assist in the form completion (see below).

*For the benefit of readers with a copy of this article that includes grayscale figures, it should be noted here that all of the software packages described herein that present Graphical User Interfaces (GUIs) to their respective user base make extensive use of color to enhance usability.

- A feedback pane, providing various notifications. For example, when automated local project validation is performed, a report indicating "No errors found" or a list of error messages containing clickable links are produced.
- A contextual help pane, providing a clickable step-by-step walkthrough of the underlying process of creating an ALMA proposal.

The process of preparing a science project for ALMA consists of two phases: Phase I and Phase II. During Phase I, the initial telescope proposal is created. Forms designed to capture the necessary high-level parameters of the proposal are completed, and various details are supplied regarding the fields to be observed, the observing mode(s) to be used, the required overall sensitivities, target frequencies, resolutions and so on. Externally-prepared PDF files containing (mandatory) scientific and technical justifications, plus any (optional) figures, tables, etc. are also added to the proposal at this stage. A sensitivity calculator is also available, as well as links to external online image and spectral line databases, online object name resolution, ephemerides of solar system objects, etc. The hierarchical tree structure of ALMA observing programs encourages the user to break down the overall project into a number of "Science Goals", which each contain a description of the science goal, the details of the field(s) to be observed, the calibration strategy to be used, the spectral set-up to be used and various other control and performance parameters. For the field setup, a GUI is available that allows the user to view images of the target fields from various online surveys, and interactively use drag and drop techniques to specify (e.g.) areas to be covered by pointed mosaics. For the spectral setup, a GUI indicating the frequency range of the ALMA receiver bands is available. This includes features indicating the Local Oscillator frequency, selected spectral line frequencies, atmospheric transmission, etc. At any point in the Phase I process, it is possible to save a copy of the ALMA proposal to disk so that the work may be retrieved and completed at a later stage. The OT also includes a validation option, permitting the PI to run an automated check on the proposal at any point to identify obvious errors or omissions, and can also generate printable summary reports of proposals for use by (e.g.) technical assessors.

When a proposal is ready for submission, the PI provides the OT with the user ID and password previously recorded in the online database list of registered ALMA users (see the description of the ALMA User Portal below). If this is in agreement with the PI user ID specified in the proposal, then the proposal is sent to the server, which then stores successful submissions in the ALMA Science Archive (ASA). Under the current system, proposals may be re-submitted and updated up to the point at which the advertised deadline is passed.

Once the proposal submission period has ended, the information stored in the submitted project is used to automatically generate Phase I Scheduling Blocks (SBs), which are then used by the APRC Simulator Tool (see below) for modelling the long-term schedule of the observing period in question. SBs are atomic sections of the proposed total observing program with a typical execution time of approximately 30 minutes that have been optimized for queue observing, and are grouped into one or more "ObsUnitSets", which denote the points during the observing program execution at which the automated ALMA pipeline data reduction is to be run.

When a project is approved for ALMA observation, then the PI is notified and the project may be retrieved from the ASA using the OT for Phase II preparation. At this point, the project will already have been flagged as being at the Phase II stage, and when retrieved will contain a first version of the Phase II Project. This Phase II project still provides read-only access to the original Phase I proposal and all its documents, and contains the Science Goals in their Phase II form. From these Goals the user will then press a button to automatically generate all of his/her Phase II SBs. Phase II SBs are the observing blocks that are to actually be executed using the ALMA array. In many cases, the initial automatically generated Phase II SBs will already be suitable for execution, but any changes to the proposal arising as a consequence of the review process (e.g. the merging of two overlapping projects) that have been made since Phase I would be implemented at this point. An expert PI may wish also discuss additional optimization of the Phase II SBs with their ARC support staff, if necessary. Once the Phase II SBs have been finalized, stored to the ASA and been approved by the supporting ARC staff, the SBs will be flagged as ready for execution.

2.2 Submission Server

The proposal submission process requires the use of a submission server to receive the final versions of the proposals. The server permits the submission of proposals only within pre-specified submission periods, enforces deadlines, runs server-side validation of all proposals at submission and assigns each proposal a unique project code. Although the submission server currently runs as a service on a Linux server, it is envisioned that by the start of full ALMA science operations, the submission server will include a control and monitoring GUI component for use by the ALMA Proposal Handling Team (PHT; a small team of staff assigned to oversee the whole proposal submission and review process). This server interface will include the ability to easily set the start and end dates for proposal submission periods, provide the ability to graphically track proposal submission numbers and oversubscription as a function of time, and so on.

3. OBOPS

Development Lead: Maurizio Chavan (ESO)

A relatively diverse set of software tools fall under the remit of the ALMA ObOps subsystem. These tools share the common feature of being both necessary for ALMA observatory operations and yet clearly not falling under the categorization of the other subsystems. The ObOps sub-projects are summarized below.

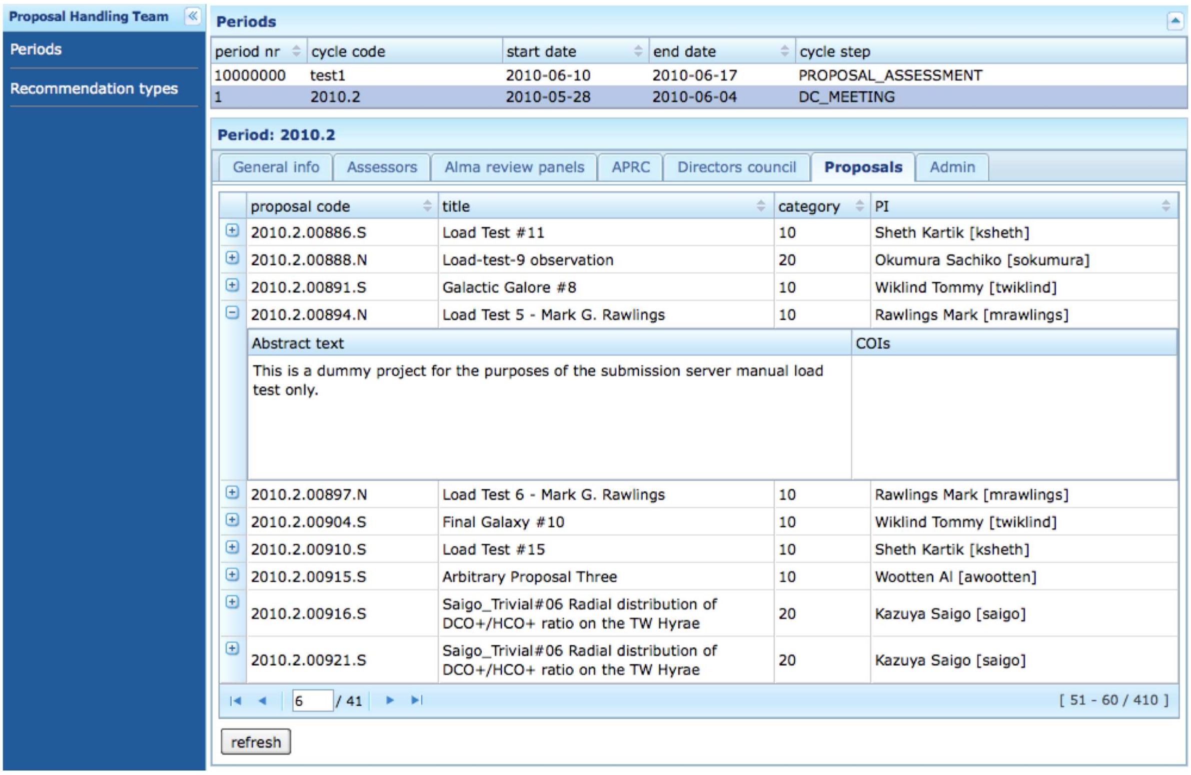
3.1 The Phase 1 Manager (Ph1M)

In summary, the ALMA proposal review process consists of three main stages. The first involves four pairs of ALMA Review Panels (ARPs), which conduct initial assessments, plus technical assessments by ALMA science staff. The second stage involves a single ALMA Proposal Review Committee (APRC), consisting of the ARP Chairs plus an APRC Chair, which consolidates the output of the ARPs and generates a single, prioritized list of all the proposals. The third and final stage involves the output from the APRC being adjusted (if necessary) by the ALMA Directors Council (ADC) to produce an observing plan for the whole schedulable period.

The Phase 1 Manager (Ph1M) is a web-based AJAX (Asynchronous JavaScript and XML) GUI tool built with the ZK toolkit. It is designed to facilitate the various tasks associated with the processing of ALMA proposals from their submission through to the conclusion of the proposal review process. The user interface makes use of widely-familiar GUI elements, such as data entry forms, drop-down menus, drag-and-drop actions, etc. Multiple parallel submission periods are supported. The Ph1M is part of the ALMA Single Sign-On (SSO) system, meaning that registered ALMA users can gain access to it and a number of other ALMA online tools with a single set of authentication credentials via the ALMA User Portal. The ALMA SSO system is based on the Central Authentication Service (CAS) project.

The Ph1M features a range of ”sub-tool” interfaces, the presentations of which are specifically tailored to the roles assigned to each user. Members of the PHT, who have overriding Ph1M administrator privileges, are able to define the times/dates for each stage of the proposal review process, assign proposals to scientific and technical assessors, and assign various functional roles to other users. An example of the user interface is shown in Fig. 2.

Each such role in the Ph1M is defined by a set of permissions. For example, a registered ALMA user may be designated as a member of the pool of assessors for a given submission period, and then be further identified as a member of one of the ARPs. This user would then be presented with a range of forms. One of these would allow him/her to indicate which assigned proposals might require reassignment due to a conflict of interest. Another such form would enable the retrieval of all proposals accepted for review. Other forms would allow the creation, interim storage and final submission of science referee reports, and allow the provision of additional comments during the ARP meeting. Similarly tailored role-appropriate sub-tools are also available for members of the APRC and ADC, and technical assessors. Various role-appropriate summary reports can also be easily generated for each stage of the proposal review process. A number of additional features specifically designed to assist the PHT are also supported by the Ph1M. Drag-and-drop assignment of members of the reviewer pool to the various panels may be performed. Furthermore, when proposal submission for an observing period has closed, and the ARP members and technical assessors have all been identified, an ”auto-assign” feature can be used to automatically initially assign proposals to each of these role holders. This process alleviates the need to



Periods

period nr	cycle code	start date	end date	cycle step
10000000	test1	2010-06-10	2010-06-17	PROPOSAL_ASSESSMENT
1	2010.2	2010-05-28	2010-06-04	DC_MEETING

Period: 2010.2

proposal code	title	category	PI
2010.2.00886.S	Load Test #11	10	Sheth Kartik [ksheth]
2010.2.00888.N	Load-test-9 observation	20	Okumura Sachiko [sokumura]
2010.2.00891.S	Galactic Galore #8	10	Wiklind Tommy [twiklind]
2010.2.00894.N	Load Test 5 - Mark G. Rawlings	10	Rawlings Mark [mrawlings]

Abstract text	COIs
This is a dummy project for the purposes of the submission server manual load test only.	

proposal code	title	category	PI
2010.2.00897.N	Load Test 6 - Mark G. Rawlings	10	Rawlings Mark [mrawlings]
2010.2.00904.S	Final Galaxy #10	10	Wiklind Tommy [twiklind]
2010.2.00910.S	Load Test #15	10	Sheth Kartik [ksheth]
2010.2.00915.S	Arbitrary Proposal Three	10	Wootten Al [awootten]
2010.2.00916.S	Saigo_Trivial#06 Radial distribution of DCO+/HCO+ ratio on the TW Hyrae	20	Kazuya Saigo [saigo]
2010.2.00921.S	Saigo_Trivial#06 Radial distribution of DCO+/HCO+ ratio on the TW Hyrae	20	Kazuya Saigo [saigo]

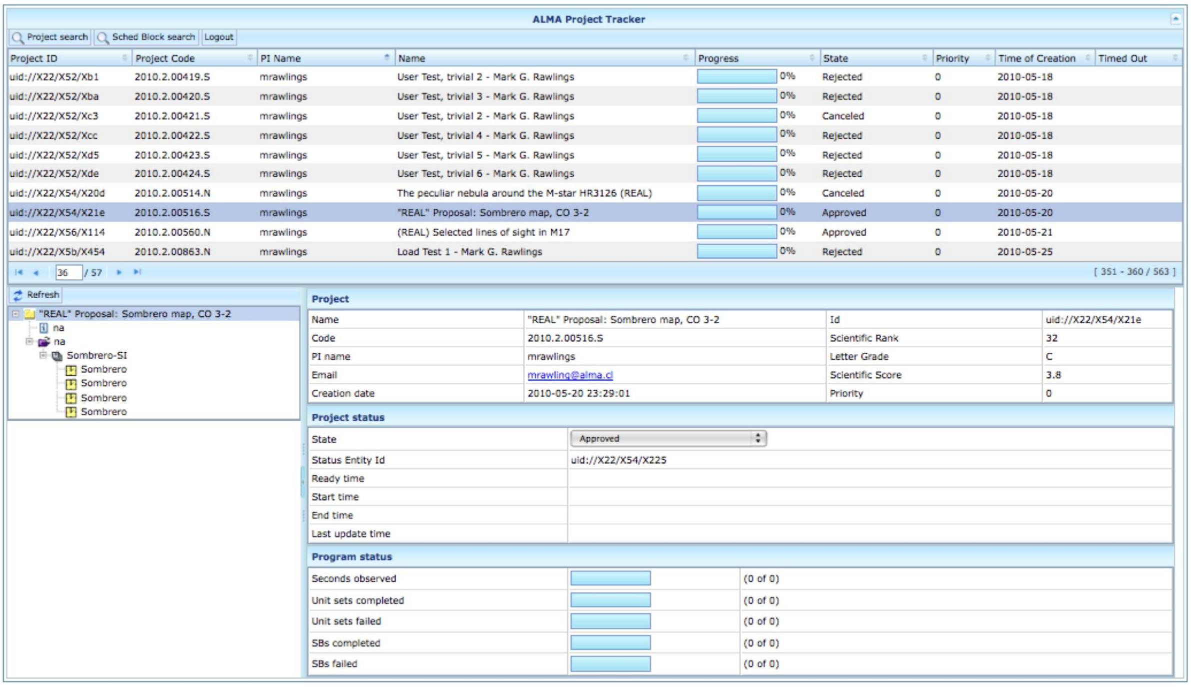
Figure 2. A screenshot of the ALMA Phase 1 Manager (Ph1M) Tool. The image shows an example of the Proposal Handling Team (PHT) interface. The upper-right panel lists the details of the created observing periods, and the lower-right panel displays a list of proposals for the chosen period. The entry for a proposal has been expanded to show the abstract text and co-I list. The left-hand panel allows the selection of additional Ph1M configuration options available to the PHT members.

manually assign a large number of proposals, and also factors in rules that avoid generating obvious conflicts of interest. Subsequent manual proposal reassignment may also be performed, should the need arise. Additionally, the PHT may use the Ph1M to monitor the return of science and technical assessments, issue e-mail reminders if necessary, and so on.

3.2 Project Tracker (ProTrack)

When ALMA science projects have been submitted to the ASA, their status needs to be tracked. This tracking must take place not just at the per-project level, but also at the ObsUnitSet and individual SB levels as well. Elements at each of these three levels may be in one of a number of possible states at any given moment. For example, a project might be identified as ”Partially Observed”, because one of its ObsUnitSets has been flagged as having been ”Fully Observed”, whereas another ObsUnitSet in the same project might contain multiple SBs, some of which are still only flagged as ”Ready” for observation. Each of these three levels making up an ALMA project must therefore follow an evolutionary lifecycle, in accordance with the flow charts presented in the internal ALMA document describing the project lifecycle in detail.[2] The transition from one such state to another may normally only be triggered by certain actors under certain circumstances. These actors may be human for some state transitions and software subsystems for others. For the large numbers of projects typical of an observatory of the size of ALMA, simply keeping track of the projects to ensure maximal completion is a significant operational task.

Another of the main pieces of ObOps software, the ALMA Project Tracker (ProTrack), has therefore been specifically designed to address this need. This is another web-based AJAX GUI tool, again using the ZK toolkit and supporting the ALMA SSO credentials authentication. The fundamental aim of ProTrack is to provide a



Project ID	Project Code	PI Name	Name	Progress	State	Priority	Time of Creation	Timed Out
uid://X22/X52/Xb1	2010.2.00419.S	mrawlings	User Test, trivial 2 - Mark G. Rawlings	0%	Rejected	0	2010-05-18	
uid://X22/X52/Xba	2010.2.00420.S	mrawlings	User Test, trivial 3 - Mark G. Rawlings	0%	Rejected	0	2010-05-18	
uid://X22/X52/Xc3	2010.2.00421.S	mrawlings	User Test, trivial 2 - Mark G. Rawlings	0%	Canceled	0	2010-05-18	
uid://X22/X52/Xcc	2010.2.00422.S	mrawlings	User Test, trivial 4 - Mark G. Rawlings	0%	Rejected	0	2010-05-18	
uid://X22/X52/Xd5	2010.2.00423.S	mrawlings	User Test, trivial 5 - Mark G. Rawlings	0%	Rejected	0	2010-05-18	
uid://X22/X52/Xde	2010.2.00424.S	mrawlings	User Test, trivial 6 - Mark G. Rawlings	0%	Rejected	0	2010-05-18	
uid://X22/X54/X20d	2010.2.00514.N	mrawlings	The peculiar nebula around the M-star HR3126 (REAL)	0%	Canceled	0	2010-05-20	
uid://X22/X54/X21e	2010.2.00516.S	mrawlings	"REAL" Proposal: Sombrero map, CO 3-2	0%	Approved	0	2010-05-20	
uid://X22/X56/X114	2010.2.00560.N	mrawlings	(REAL) Selected lines of sight in M17	0%	Approved	0	2010-05-21	
uid://X22/X5b/X454	2010.2.00863.N	mrawlings	Load Test 1 - Mark G. Rawlings	0%	Rejected	0	2010-05-25	

36 / 57 [351 - 360 / 563]

- "REAL" Proposal: Sombrero map, CO 3-2
  - na
  - na
    - Sombrero-SI
      - Sombrero
      - Sombrero
      - Sombrero
      - Sombrero

Project

Name	"REAL" Proposal: Sombrero map, CO 3-2	Id	uid://X22/X54/X21e
Code	2010.2.00516.S	Scientific Rank	32
PI name	mrawlings	Letter Grade	C
Email	mrawling@alma.cl	Scientific Score	3.8
Creation date	2010-05-20 23:29:01	Priority	0

Project status

State	Approved
Status Entity Id	uid://X22/X54/X225
Ready time	
Start time	
End time	
Last update time	

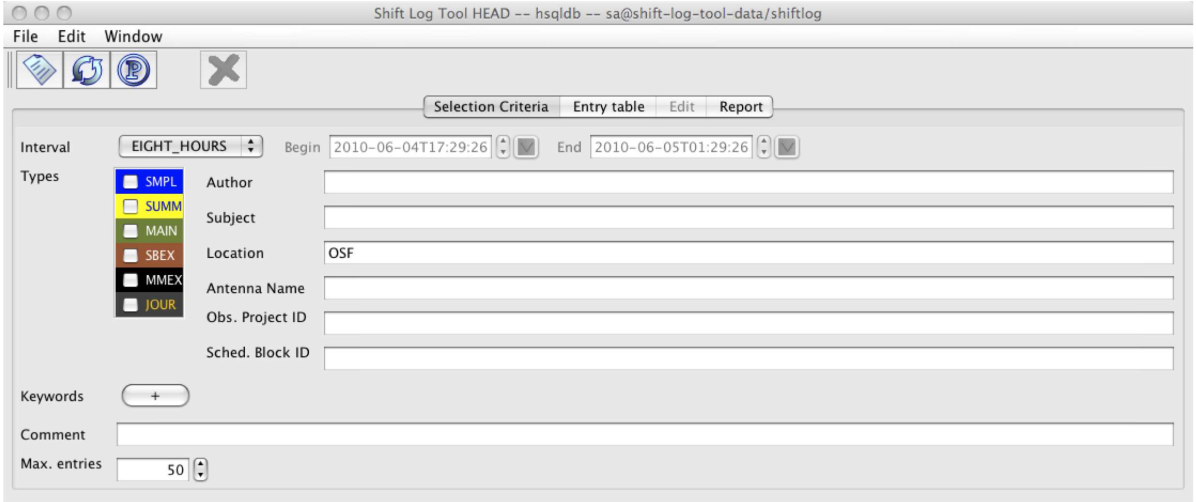
Program status

Seconds observed		(0 of 0)
Unit sets completed		(0 of 0)
Unit sets failed		(0 of 0)
SBs completed		(0 of 0)
SBs failed		(0 of 0)

Figure 3. A screenshot of the ALMA Project Tracker (ProTrack) Tool. The image shows an example of a project search result. The upper pane shows the list of proposals produced by the search. The lower left pane shows the tree structure of the selected proposal in the search results pane. The lower-right panes show the details of the selected part of the project tree.

means of monitoring and (where necessary) adjusting the status of the various components of all of the ALMA science proposals/projects. As for the Ph1M, the ability of each user to view and/or modify its content is governed by user roles.

The ProTrack GUI offers a suitably-authorised user the option to search the ASA for the projects (or SBs) according to a number of criteria. These criteria include obvious fields such as ”PI Name”, ”Project Code”, and so on, as well as by a number of additional criteria based on the status of the project or SB at that given moment. For example, it is possible to perform searches for projects that have reached an estimated fraction of completion, SBs that have particular state (e.g. ”Approved” by an ALMA staff astronomer), SBs that have been observed between a specific set of dates and so on. The results of such searches are presented as a list of hyperlinked records, the selection of which produces additional project- or SB-specific content in panes below it. A pane to the lower left displays an expandable tree structure analogous to that of the OT. Selection of a particular program component such as an SB, fills the panel to its right with summary information on that component's status. An example of the user interface is shown in Fig. 3. The information displayed here may include the abstract text of the project, the completion status of each component, and so on. This interface will allow project PIs to log in to the system and see the progress to date on their projects. Depending on the permissions associated with the role of the user, the state of some project components may also be changed using drop-down menus containing options in accordance with the transition rules of the project, ObsUnitSet or SB lifecycle. ProTrack will also interact with the automatic Scheduler and Quality Assurance (QA) software (see below), in order to ensure that the correct state is always recorded for each SB, ObsUnitSet and project at every stage of the science project lifecycle. Lastly, ProTrack will also be able to generate a number of reports that will be needed for routine observatory operations. These reports will include graphical plots of parameters such as the project completion for a given observing period, numbers of projects of a given grade still awaiting observation, etc.



Shift Log Tool HEAD -- hsqldb -- sa@shift-log-tool-data/shiftlog

Interval EIGHT_HOURS Begin 2010-06-04T17:29:26 End 2010-06-05T01:29:26

Types

Author

Subject

Location OSF

Antenna Name

Obs. Project ID

Sched. Block ID

Keywords

Comment

Max. entries 50

Figure 4. A screenshot of the standalone version of the ALMA Shift Log Tool (SLT).

3.3 Shift Log Tool (SLT)

The Shiftlog Tool (SLT) is an ObOps software tool that allows ALMA observing staff to record a log of observing activities and other notable information during actual ALMA observing shifts. The tool will ultimately offer two methods of interaction with observatory logs. The first of these is a standalone Java-based GUI utility that is deployed as part of the standard bulk software releases deployed at the ALMA sites, and the second is a password-protected web-based interface that is still under development, essentially offering the same functionality for authorized remote users. An example of the user interface of the standalone tool is shown in Fig. 4. All shift log comments are searchable via either interface, and the logs searches can be conducted on the basis of a number of parameters, such as specified logging time intervals, comment author, subject, filing location, individual antenna name and project or SB identification code. Comments may also be assigned keywords. Several categories of user comments are supported, such as simple routine comments, shift summaries, maintenance and science journal entries. Standard organizational templates for these entries are supported. As well as user-created comments, the shift logs generated also include automated entries triggered by routine events, such as the commencement of observation of a new SB. Manual shift log comments are entered as text, but may also include the attachment of several types of other files as well (e.g. images). Overall shift log reports can also be generated and exported as either plain text files or as HTML documents.

3.4 Other ObOps subsystems

In addition to the above tools, a number of other utilities and considerations also fall under the purview of ObOps. These include consideration of the project lifecycles (discuss above) and the implementation and deployment of the SSO system. In addition, ObOps will be providing a Data Packer tool, which will automate the generation of the final data packages that will be delivered to the project PIs, and several additional GUIs to simplify the examination and configuration of the necessarily large number of system monitor points. Bespoke software for QA and Trend Analysis are also under development as part of the ObOpbs subsystem, and are explicitly discussed in a separate section below.

4. SCHEDULING

Development Lead: Rafael Hiriart (NRAO)

Once proposals have passed the review process and their Phase II SBs have been generated, stored in the ASA and approved for observation, the actual observations themselves still need to be scheduled for execution. The Scheduling subsystem therefore explicitly addresses this point. Successful observing projects, as well as observatory tasks broken down into SBs will be residing in the Long Term Queue (LTQ). Short-term scheduling will be performed by scheduling software that will select the next SB to be executed out of the whole pool of SBs available for the observing period, based on a number of factors.

4.1 Short-term Scheduling

There are many factors that need to be taken into account when deciding on the execution of a specific SB of an observing project. These factors can be broadly classified into the following three categories:

- **Project Factors:** These include the final proposal grade/ranking, any interdependencies between SBs (e.g. the need for one particular SB to be successfully completed before another particular one is to be started), the degree of completion of each project (and ObsUnitSet), the overall ALMA time shares available to each Executive (North America, Europe East Asia and Chile) and any particular time limitations on each project. Specific project requirements also need to be considered, and these include the required final signal-to-noise ratio needed, phase stability criteria, spatial resolution/*uv*-coverage needs, calibration accuracy, pointing accuracy, calibrators, front-end/back-end configurations, sub-array requirements, short-spacing data needs, and the possible need to pause a project at key points to evaluate the data acquired so far.
- **Environmental Factors:** These include weather conditions (opacity, wind, the system temperature, atmospheric turbulence effects on phase stability, etc.) and the predicted behaviour of the weather during the expected execution time period of the SB.
- **Configuration Factors:** These include hardware availability issues (antennas, receivers, correlators, back-end, etc), the current array configuration, the availability of the requested observing modes, the observing source availability (i.e. the current local sidereal time), and antenna shadowing effects.

In addition to the above considerations, the short-term scheduling software must also factor in the need to schedule specific maintenance tasks and the execution of observatory tasks that require actual observations, such as the monitoring of array parameters that slowly vary over a period of several days. Target of Opportunity (ToO) and Director's Discretionary Time (DDT) project scheduling also needs to be accommodated.

At any given moment, all the SBs that have not yet been executed will be accessed by the short-term scheduling software from the archive. They will be automatically prioritized, on the basis of the above parameters (with appropriate weighting functions) and a list of the top $10-20$ produced. The grounds for the selection of these SBs will be displayed on the screen and fully documented by the logging capabilities of the software. Since, on average, a single SB will have a duration of approximately 30 minutes, and it can typically take up to 15 minutes to ready a freshly-selected ALMA Band for observations (e.g. if the FE cartridge is not already powered up), the algorithm will have to compute the prioritized list $15-20$ minutes ahead of execution. Furthermore, since it is expected that as many as ten thousand SBs may be present in the pool, the algorithm must be rapid enough to avoid delaying observations, and it must also have sufficient buffer space for any intermediate operations that are needed. When generating the prioritized shortlists, Scheduling must also have access to current weather parameters, in order to incorporate a prediction of how the weather will evolve on timescales of a few hours, and the database of current configurations and available observing modes. During full science operations at ALMA, it is expected that the algorithm will be mature enough to make very few mistakes in its choices, and by default, the Scheduler will be permitted to proceed automatically with the execution of the top SB in the prioritized list it has computed. The option for the manual execution of a particular SB will, however, still be retained. For the purposes of improving the algorithm performance, any such overriding decision made by Department of Science Operations (DSO) personnel will act as weighted feedback into the system, which will keep optimizing the parameters used in all of the subsequent SB execution decisions. Whenever an SB has been executed and has run to completion, initial Quality Assurance will be conducted by the ALMA Astronomer on Duty (AoD). The SB can then be either deemed finished (and flagged accordingly as part of the ProTrack-monitored project lifecycle summarized earlier) or sent back to the queue for re-execution if deemed unsatisfactory (again, as part of the project lifecycle).

The algorithm to be used in the grading of the SBs by the Scheduler is currently still being designed. However, it has already been recognized that for convenience and efficiency the algorithm will need to consist of two parts. The first of these will deal with the "binary" parameters that just require a yes/no decision. To those SBs that clear this first stage, a second round of tests will be applied, consisting of a weighted formula with adjustable parameters. Once all the SBs have been graded by the Scheduler, they will be sorted and the list will be produced and displayed as output. The algorithm will be trained for optimization during the initial 12-month Early Science period, with the decisions of the AoDs being compared with the recommendations of the algorithm to generate feedback that will be used to further optimize the parameters (Bayesian and/or neural network training algorithms are being considered for this). Tests of the optimization will then be performed by re-running the scheduling algorithm on the historical SB execution data.

4.2 Long-term Scheduling

In order to enable the creation of an optimal pool of SBs for an observing period, it is essential for the APRC to be able to make informed decisions during the meeting about the final proposal ranking. This therefore ideally requires some knowledge of the implications of any potential proposed ranking scheme. To address this issue, an additional tool, the APRC Simulator Tool, is being developed by the Scheduling subsystem team. This tool will work with Phase I SBs to provide the APRC with projections of the science time usage of ALMA during the upcoming observing period. The SBs that are to be used as input for these projections are automatically generated from the Phase I proposal submissions stored in the ASA, with only the information relevant to long-term scheduling from the SBs being retrieved by the APRC Simulator Tool. Any constraints in terms of hardware configurations, etc. for the observing period in question will also be taken into account in the simulations, and the tool will also incorporate Chajnantor weather profiles for several scenarios (average year, good year, bad year) with a time granularity shorter than the expected average SB duration of 30 minutes. Modelling constraints arising from the accepted percentages of observing time allotted to each of the Executives will also be incorporated. in terms of output, the tool will allow the APRC some latitude for experimentation with (e.g.) adjustment of the rankings and/or divisions between the proposal grades during the meeting. This will all be presented to the APRC members via a GUI, enabling them to more easily identify possible crowding of specific LST ranges, antenna configurations, etc., and generate output in the form of both text and graphical plots. The same tool will also be used by the ALMA System Astronomers to optimize the array configuration deployment.

5. ARCHIVE

Development Lead: Andreas Wicenec (ESO)

In its final form, the ALMA archive will be a fully distributed database system, with operational parts at all ALMA sites, including the ALMA Regional Centres (ARCs), the only major exception being the Array Operations Site (AOS). It will provide storage and internal and worldwide query and retrieval interfaces for both engineering and scientific data, the latter of which will be Virtual Observatory (VO)-compliant. The ALMA Archive Subsystem group is responsible for its development and initial deployment.

The ALMA Archive Subsystem is central the ALMA computing infrastructure and provides generic persistence and the main data flow mechanisms for data delivery from the Operations Support Facility (OSF) to the Santiago Central office (SCO) and to the ARCs. In addition to these data flow support functions, the archive is also responsible for the long-term maintainability of the ALMA data even beyond the lifetime of the observatory. The Archive subsystem is therefore one of the most critical subsystems, dealing with all ALMA meta-data, logging and bulk data. The archive will ultimately be a distributed system that needs to be operated in a concerted, efficient and secure way at the different sites in order to be able to deal with the data rate, the total data volume and the internal and external data requests of ALMA and the worldwide astronomical community. The archive is also essential to ALMA operations: without a functioning archive, the computing control systems will not start, and if the archive fails or the connection to the archive is lost for some reason, ALMA will have to stop observing almost immediately.

In terms of design, the ALMA archive is divided into two major parts: the ALMA Science Archive (ASA) and the ALMA Frontend Archive (AFA). This structure will also be replicated to each of the ARCs. The AFA is the part of the archive providing the core persistence functionality for the ALMA data. The AFA also provides the software interfaces to the other subsystems, plus engineering and lower-level scientific interfaces for internal human users. The ASA provides the external interfaces to scientists and VO systems. It also implements the science user's view of the ALMA data. For managerial and security reasons, not all user gateways will actually be enabled on each of the sites. For instance, the QA and engineering interfaces will only be enabled at the SCO, whereas the gateway for general science user access will be enabled at the ARCs. The strict separation between the ASA and the AFA enhances protection of the AFA content and ALMA operations while still permitting independent changes to be made to the metadata items.

The ALMA archive design reflects the three main categories of data produced by the observatory: bulk data (generated by the correlators), XML data (project and science metadata) and monitor and log data (in general, timestamped values). The main components of the archive are therefore as follows:

- An XML store for the storage of project- and observation-related metadata in an XML database.
- A bulk store with a New generation Archive System (NGAS) backend for the storage of large binary data on a file system.
- A monitor store for the storage of sensor data and logging data, logically grouped in a relational database.
- The science archive. This handles the storage and maintenance of scientific metadata in a database. All data products produced by the standard pipeline will be incorporated. These include calibrated images (data cubes), data reduction and imaging scripts, Quality Assurance data and associated parameters, environmental data, and observing proposals, complete with SBs.
- A database service that will provide persistence for the whole archive. This is the core component of the whole ALMA Archive subsystem.

The current ALMA Archive design allows for a maximum data rate of $\sim$ 64 MB/s and an average data rate of $\sim$ 6.4 MB/s. The maximum data rate is limited by the maximum speed of data ingestion into the archive, and the average data rate yields the necessary long-term storage capacity of approximately 200 TB/year. The ALMA correlators are capable of producing a much higher data rate than this maximum (up to $\sim$ 1000 MB/s), especially when a project calls for the maximal spectral channels and short dump times, and the archive has been designed to be upgradable to address such future potential demands.

Data are transferred from the AOS down to the OSF through a fibre link. Bulk and XML data in the ALMA Project Data Model (APDM) and ALMA Science Data Model (ASDM) formats then flow from the OSF to the SCO network by network. The network link between the OSF and SCO must have a transfer rate of at least 150 Mbit/s to sustain the average data rate plus maximum data rates during limited periods of time. It must also be possible to operate ALMA even if the OSF - SCO link is broken, as the overall network structure means that the OSF archive is a potential single point of failure for ALMA data acquisition, This therefore requires that the OSF archive is required to be a high-availability system. The NGAS installation at the OSF also has the highest requirements in terms of availability and throughput. Since there is no large high-speed buffer between the correlators and the NGAS system, the front-end cluster of NGAS nodes is required to be able to archive an approximate sustained rate of 66 MB/s.

The SCO will hold the full operational reference copy of the ALMA data and will host the science archive. It will provide and maintain the main interfaces to the ARCs and internal archive users and the pipeline. This requires that there be a full archive installation, including the database, software and NGAS clusters. The availability requirements are similar to those of the OSF, because most of the data processing, content management ands science archive construction will take place at the SCO.

Data are then replicated from the SCO archives to the ARC mirror archives through a network connection (XML data) and hard disks or network (bulk data). Pipeline products generated at the SCO (see below) are also replicated to the ARC archives. Science proposals submitted using the OT enter the SCO archive and are subsequently replicated to the ARC and OSF archives, as are any associated SBs. External ALMA end users will receive all data from the ARCs.

During Science Operations, the ARC archives may be synchronized with the central SCO archive on two different timescales. Small information sets (e.g. proposal and observation preparation information, science pipeline images) shall be immediately replicated to the ARC nodes via an internet link. Larger data sets (e.g. unprocessed correlated *uv* data, engineering data streams) will be moved via physical media (probably hard-disks) initially, and subsequently via the internet, subject to future cost and reliability.

6. PIPELINE

Development Lead: Lindsey Davis (NRAO)

Once ALMA science data acquisition has commenced, those data need to be reduced. ALMA data reduction takes two basic forms. The first of these involves the initial inspection of the data during (or immediately following) the observations, to ensure that ALMA is functioning normally and that the data do not exhibit major problems, and the second is the subsequent detailed pipeline reduction that produces the final, fully-reduced science data products and other associated data.

6.1 Quicklook

The Quicklook software package is designed to help DSO personnel (primarily the AoDs) assess the quality of the data and the overall performance of the array during the execution of a single SB (or during the consecutive repetition of the same SB). It will therefore be principally used to determine metrics for the initial Quality Assurance (QA0) parameters.

The two main purposes of Quicklook are:

- the display of TelCal results (TelCal being the software handling the real-time reduction of calibration observations);
- the online reduction of the data just taken for a single SB (or consecutive repeats of the same SB).

Within an SB, there will be observing scans flagged with a Calibration Intent that trigger a TelCal reduction once completed. TelCal is capable of processing all types of calibration observations during normal synthesis/mosaicing and single-dish observations (i.e. pointing, focus, amplitude and phase, antenna-based gains, WVR correction, polarization, bandpass, etc). A full discussion of TelCal is beyond the scope of this paper. Once the calibration has been reduced, TelCal broadcasts this fact and creates an ASDM auxiliary table with the results and writes it to the archive. The Quicklook program, when triggered by the TelCal broadcasts, fetches the calibration results from the archive and displays them as a function of time, baseline or antenna as required. Associated with these calibrations are any of several alarms intended to notify the observing staff of any sub-optimal results obtained for specific baselines and/or antennas. Distributed display capabilities are also planned to allow the running of similar sessions on several computers simultaneously. Automated summaries of the calibrations will be produced and archived for future reference by DSO personnel (mainly for trend analysis use by the System Astronomers).

The so-called Array Monitoring capability of Quicklook allows the display of the current *uv*- coverage (a “snapshot” view), the integrated *uv*-coverage over the whole (or repeated) SB(s), simple imaging deconvolution (including mosaicing) and single-dish observing modes to check the quality of the data. The reduction will be based on a subset of the CASA scripts used for the Science Pipeline (see below), including some of the heuristics. Since this reduction is intended only as a quick check for detections, quality of the “dirty” beams, noise level checks, etc, it is envisioned that it will not include the merging of single-dish (zero-spacing) and interferometric data, and will, in general, be limited to standard observing modes with a minimal choice of reduction parameters. This mode of Quicklook will be automatically configured and run in the background, and will not generally require real-time user input in order to set up parameters. The observing staff may, however, adjust Quicklook options to display different parameters, or to zoom in on certain details/regions of specific plots.

6.2 Science Pipeline

The purpose of the Science Pipeline is the reduction ALMA data in an automated fashion using standardized CASA scripts appropriate for each of the ALMA observing modes. The need for an automated reduction arises both from the requirement to serve novice ALMA users and the many available possibilities in terms of observing modes, correlator configurations, etc. During full science operations, it is anticipated that the Science Pipeline will produce reduced data of publishable quality, and that only expert users that wish to explore alternative reduction strategies will opt for a re-reduction of their datasets. To achieve this goal, the parameters in the CASA reduction scripts that will be used by the Science Pipeline need to be optimized for each standard observing mode. This task has been undertaken by the Pipeline Heuristics Group which, by using data from both ALMA and other arrays, will work on refining the incorporation of all of the various input parameters into these scripts. The resultant reduction processes will also generate some third-stage QA information (QA2; see below) that can be used to assess the quality of the final data product.

The bulk of the initial data reduction will be conducted using the Science Pipeline at the SCO. This will first require that the data be transferred from the provisional archive at the OSF to the main archive at the SCO. It is currently envisioned that there will be one full Science Pipeline installation at the SCO and one at each of the ARCs, i.e. four in total. All of these will have the same architectures and run the same software, but the ARC Pipeline clusters will be used for the testing of new software releases and the re-reduction of some selected datasets. The ARCs will also be responsible for offline Pipeline operations, such as the handling of external user requests to extract and reduce older (non-proprietary) datasets from the archive. Re-reductions may be requested by the PI, but these will take place at the ARCs to avoid adding to the SCO Pipeline workload. Science Pipeline reduction will be optimized in order to take full advantage of parallelization. Even so, current simulations of projected ALMA data acquisition and reduction rates indicate that it may be necessary to run the Science Pipeline continuously during full science operations. The Pipeline has already undergone initial testing with real ALMA single dish data in Chile.

The operation of the Science Pipeline will be event driven, either being triggered by the Pipeline Operator or automated via a queue. PIs will be offered a few choices in the way the data will be be reduced (this will be mostly just in terms of the choices of deconvolution algorithms, and in the sizes and binning of the output cubes). These choices will be made by the PIs at the time that the project is split into SBs by the PI using the OT (Phase II). In general, however, the number of choices will be kept to a minimum in order to maintain overall data consistency within the archive.

There are currently two possible project conditions identified that will trigger the reduction of the data taken up to a given point point: the end of the execution of an ObsUnitSet and the completion of a whole project. In either case, the Science Pipeline will access all of the data associated with a given project and reduce it in a single batch. Such a batch reduction would include all of the necessary calibration files and interferometric and single-dish science data (possibly taken with different array configurations). All of the data pertaining to a project will be reduced when the project is deemed complete, irrespective of prior partial reductions that have already occurred for the project. All of the scripts used in the reduction, as well as all output log files and the reduced data, will be delivered to the PI.

A standard reduction process will involve the following main steps:

- Data Query: The archive is queried for all data relevant to a specific project;
- Data Read: All relevant data are read into the Science Pipeline;
- Reduction: The data for calibrators and targets are tagged appropriately, so that they can be identified during reduction. The most general data reduction case would be for that of a project including several pointings (mosaicing) of the same source, data taken with the ACA and the 12m Array and also including several array configurations. For such a project, the interferometric and single-dish datasets would be reduced separately first, then merged. Deconvolution would be performed and QA2 parameter evaluation conducted.

- Data Write: Complete logs of all the Science Pipeline outputs are written to the archive, together with the scripts used in the data reduction. The final output of the reduction process are image cubes with dimensions and spectral binning as specified by the PI.

When a new version of the Pipeline software is deployed at the ALMA sites, it may occasionally be necessary to re-reduce some data. Under such circumstances, the affected PIs would be notified, the re-reduction of the data in question would be performed at the ARCs and the reduced data copied back into the SCO archive.

7. ALMA QUALITY ASSURANCE AND TREND ANALYSIS (AQUA)

Development Lead: Maurizio Chavan (ESO).[†]

The quality of the ALMA data products to be delivered to the PIs will be checked by DSO Personnel at different stages as defined in the ALMA Operations Plan.[3] These stages have been identified as follows:

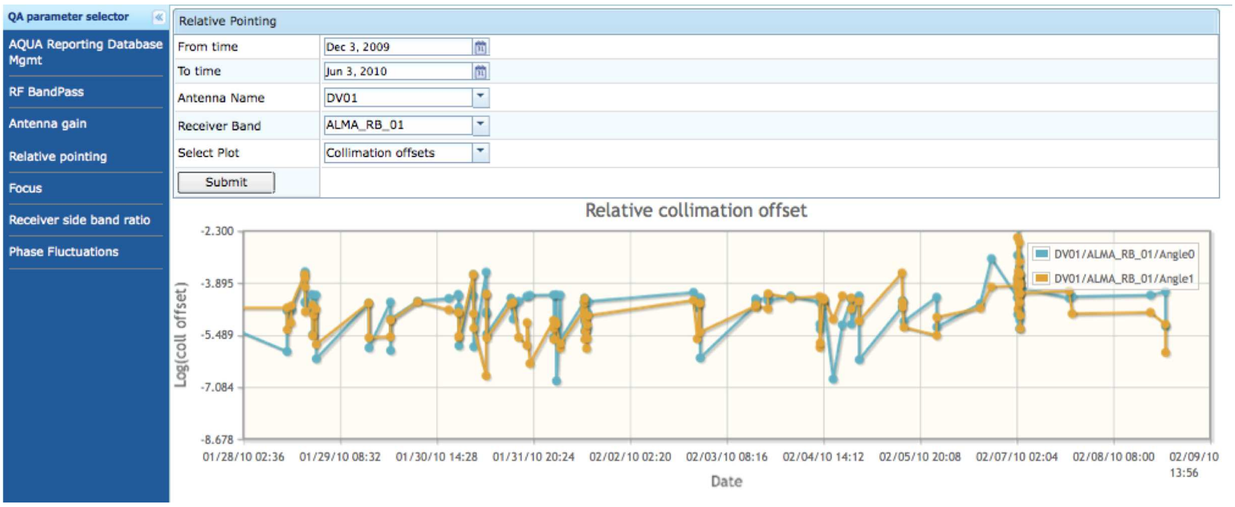
- QA0: Data Acquisition
- QA1: Observatory-Task Quality Assurance
- QA2: Data Reduction

These stages are summarized below. At the time of writing, the acceptable ranges for each of the QA parameters/metrics are still being established.

QA0 parameters deal with rapidly varying performance parameters (on timescales of the order of a typical SB execution time or shorter). QA0 thus has to be performed at the time of data acquisition QA0 will consist of real-time/semi-real-time monitoring of calibration data during its acquisition at the telescope and the calibration summaries at the end of an SB. In addition, input from Monitor and Control tools will be included. Generally, QA0 metrics/parameters have been chosen in order to allow the monitoring of the integrity of the whole signal path, from the atmosphere down to the back-ends. QA0 parameters monitor possible atmospheric effects, antenna issues, front-end and back-end issues and connectivity issues (such as delays measurements and total power levels). QA0 parameters will be used by the AoD to decide whether a given dataset has been obtained under satisfactory conditions or has to be re-observed. Given the fact that some of the QA0 parameters will result in a fairly smooth degradation of the datasets, on top of a range of optimum parameter values, an additional range for which the data still could be accepted will also be established. In addition to these default criteria for acceptability, a PI may make a technical case in the proposal for some of the QA0 parameters to be within a stricter or broader range than those normally used. A summary of the QA0 parameters will be created automatically by Quicklook and will be attached to the data packages to the PIs for reference. These summaries will be also used by the System Astronomers for trend analysis of the array performance and for future optimization of the observing/calibration procedures.

The QA1 calibration measurements required for optimum operation of the ALMA array are characterized by parameters that vary slowly with time (typically $>$ 1 week). They can therefore be loosely scheduled periodically as "observatory tasks" during a block of a few consecutive days and executed as soon as the weather is of sufficient quality. All of these calibrations will be reduced with specific software that produces a quality assessment of the fits/results. After revision of the quality metrics, the measurements will be implemented if acceptable. Measurements that are outside the optimum range may be accepted for those parameters that smoothly degrade the performance of the array. Re-measurements of any of these parameters may be triggered at any time by observations revealing significant degradation of the data quality due to that parameter. In general, if the values of these slowly-varying parameters are within the specified ranges, no additional steps need to be taken during data reduction. The measurements that have been identified as constituting QA1 include all-sky antenna pointing models, antenna focus and gain versus elevation models, baseline measurements, signal-path delay measurements

[†] The ALMA QA and trend analysis software are actually part of the ObOps subsystem, but are discussed here in a separate section in order to better reflect the overall process undergone by each ALMA project.



Relative Pointing

From time	Dec 3, 2009
To time	Jun 3, 2010
Antenna Name	DV01
Receiver Band	ALMA_RB_01
Select Plot	Collimation offsets

Figure 5. A screenshot of the ALMA QUality Assurance (AQUA) tool.

(from the front-ends down to the correlator), primary beam measurements, optics calibrations and flux calibrator observations.

QA2 addresses issues that only surface at the time of full science-grade data reduction. The Science Pipeline must merge data, taken with independent calibrations, from different arrays, and configurations. Furthermore, once merged, the deconvolution into the image plane uses highly non-linear algorithms that are compromises based on the expected dominant structural components (i.e., compact versus extended) of the target sources. It is only at this stage that the output can be compared with the project goals in terms of signal-to-noise ratio, etc. All of the required QA2 parameters will be computed by the Science Pipeline and inspected by DSO staff before the data are made available to the PIs. Although a number of the QA2 parameter metrics are easy to define, since this stage of QA is rather more abstract, some of the parameters, such as Image Fidelity clearly require several metrics and will required further study. In some projects, the requirements for QA2 may be different from the ALMA standard, and knowledge of the science requirements will be necessary for assessment. All data that pass QA2 will be released to the PIs. If it is deemed that an improvement of the results could be achieved by a re-reduction of the data with non-standard scripts, then the option of data re-reduction via the ARC Pipelines may be possible. It is expected that DSO System Astronomers will regularly check the QA2 outputs for trend analysis and optimization of the Array performance.

Trend Analysis is a broad term used to describe the data mining activities that the DSO System Astronomers will perform on the metrics described above. It covers all aspects of the performance of the ALMA Array and is directed towards improving some aspects of Science Operations. For example, in terms of array configurations, a trend analysis of the weather patterns and proposal history, will allow better allocation of time for specific configurations. Studies of the data quality obtained versus the environmental parameters monitored will be used to optimize scheduling parameters and (e.g.) refine execution time estimates for projects. Calibration observations will be also studied for optimization. There will also be detailed studies of the performance of the Science Pipeline as a function of data reduction script parameters, observing modes, etc. in order to improve parallelization and reduction strategies.

In order to monitor and the aforementioned QA metrics and parameters and effectively perform trend analysis, a software tool called the ALMA QUality Analysis tool (AQUA) is currently under development by the ObOps subsystem software development team. This is another web-based AJAX GUI tool, again using the ZK toolkit and will support the ALMA SSO credentials authentication. AQUA will enable the DSO staff to easily extract and plot the various metrics needed to perform QA. AQUA allows the specification of time periods, antennas (or arrays or baselines), receiver bands, etc. and uses them as parameters for the generation of various plots and automatic reports. Fig. 5 shows a screenshot of an early AQUA prototype. Although the tool is still in the early stages of development, it already makes extensive use of GUI elements such as user-specified zoom windows on the available metric plots.

8. OTHER USER SUPPORT SUBSYSTEMS: WEB PAGES AND USER PORTAL

Several other non-real-time software components are needed in order for ALMA to function as a working observatory. A set of DSO-hosted web pages are currently under construction, and these will contain large amounts of detailed documentation, tools and other content that will be useful to the ALMA community as a whole. The web pages will also provide access to the User Portal. This will be a Plone-based system that will support the registration and SSO role-governed login of ALMA users, acting as a gateway to the various tools described above. Offline ALMA data reduction will be available via the CASA software package. A user Helpdesk system based on a commercial product (*Kayako* is also being developed for use at the ARCs.

9. SUMMARY

In addition to the real-time telescope control systems, ALMA will make extensive use of a large number of interconnected non-real-time software subsystems. An overview of these non-real-time subsystems has been presented from the DSO perspective, covering all the major stages of an ALMA observing project, from proposal preparation through to the delivery of the resultant dataset to the PI.

ACKNOWLEDGMENTS

The authors wish to thank several members of the ALMA software development team for their helpful comments. In particular, thanks go to Alan Bridger for his comments on the ObsPrep section of the text, Maurizio Chavan for his input on the ObOps section and Brian Glendenning for his overall inspection of the paper and additional notes. Thanks also go to the other co-authors of the ALMA DSO Implementation Plan, upon which much of the text is based, and the JAO management for enabling the public presentation of this paper.

REFERENCES

1. L.-A. Nyman, M. Rawlings, B. V. Vilaro, J. Cortes, W. Dent, and R. Kneissl, “Department of science operations implementation plan.” Version A0 draft, ALMA internal document, July 2009.
2. A. M. Chavan, “Life-cycle of observing projects.” Version A draft, ALMA internal document, February 2009.
3. L.-A. Nyman *et al.*, “Operations plan.” Version E draft, ALMA internal document, March 2010.