

ALMA Operations Plan: Issues and Recommendations

Neal J. Evans II, Roy Booth, Leo Bronfman, Yasuo Fukui, Mark Gurwell,
John Richer, Seiichi Sakamoto, Peter Shaver, Christine Wilson, Malcolm Walmsley

August 30, 2001 04: 22 p.m.

Abstract

This document is intended to raise the issues and, where possible, provide recommendations to the ASAC regarding ALMA operations. We divided our considerations into two main aspects: before and during the observations; after the observations. In the first area (§1–6), a sub-group led by Christine Wilson considered the issues. Yasuo Fukui led the development in the second area, which covers issues of the regional data centers, archives, and proprietary periods (§7–9). At the end of some sections, our recommendations or topics for further discussion appear in italics. We conclude with a summary restatement of the recommendations (§10) and topics for further discussion or study (§11).

1 Proposing to Use ALMA

Preparing the Phase I proposal will lead to the first encounter with the ALMA operational system that most users will have. We believe that it is particularly important that this encounter be as welcoming as possible so that astronomers unused to radio interferometry are encouraged to observe with ALMA. It is also essential that the refereeing process be clear and informed by knowledge of what has been done and what is possible. While the details of time allocation remain to be worked out, we focus on conditions that we believe should be met by whatever method is eventually adopted. In particular, we consider the following areas.

1. Access to information about completed and currently scheduled projects.
2. Tools for preparation of both Phase I and Phase II proposals, such as time estimators, ALMA simulators, etc.
3. A process for technical review, including a quantitative measure of the stringency of the requirements.

The simulation tools have relevance to Phase I, Phase II, and operations during observing, and so are discussed in detail in the next section.

1.1 Phase I Proposal

The first step in planning any observing proposal is to assess what has already been done. For data past the proprietary period, the ALMA archive should provide this information. However, there will be many observations that are not yet in the archive, especially in the early years of ALMA. We believe that a more limited archive of information about completed proposals and a still more limited archive of information about approved, but uncompleted, projects should be available. For completed projects still in the proprietary period, a prospective user should be able to learn the names of the proposers, the coordinates covered, the source names, the frequencies, and the rms achieved in the pipeline reduction. For approved proposals, all the same information should be available, but the rms noise or integration time approved by the review panel should be supplied. The frequency information should include rest frequency and

either velocity or redshift. This information should be in an easily searchable data base that could be in the regular ALMA archive or in a separate data base.

The tools for proposal preparation should be easy to use and yet powerful. On the simplest level, we agree with the SSR report that the whole proposal process should be electronic. It should be possible to upload proposals to the proposal data base, but also to download one's own proposals, modify them, and upload them again, up to the deadline time. The proposal should include enough information for a scientific and technical review, as well as enough information to allow automated checking of the final observing scripts against the parameters of the approved proposal. This will mean that a detailed source list is required, including information on coordinates, frequency and field of view. There will have to be exceptions or special procedures for time-variable sources or targets of opportunity. In addition, in the case of very long source lists, particularly those which are easily characterized by other means, the observer should be able to apply for and justify an exception to the detailed source list rule for Phase I proposals.

Recommendation: Complete information on the source parameters (coordinates, velocity, frequency, resolution, rms noise) in approved and completed projects (both proprietary and public) be available in the archive.

1.2 Phase II Proposal

The main goal in the Phase II proposal is to create appropriate Scheduling Blocks that realize the written scope of the successful proposal. One issue is how to get advice from an expert if it is needed, and where that expert should be located. From the user's point of view, it would be useful to have expert advice located in a similar time zone (i.e. experts located at a single RDC would probably suffice for astronomers in North America); it might also be important to be able to get expert advice in the native language (even if Phase I and II Proposals are all in English). However, as long as the advice is readily available when it is needed, it could be possible to have all the experts located in Chile, if that was the decision of the project. The amount of human interaction in Phase II will probably be higher in the early stages of ALMA and settle down to some lower level as the project matures. However, there will likely always be some need for expert advice in Phase II preparation from beginning observers or from observers wishing to develop complex programs.

1.3 Calibrations

The key question here is what is the responsibility of the project and what is the responsibility of the observer. Since we want ALMA to be accessible to non-experts, some basic calibration responsibilities need to be accepted by the project. For example, perhaps the Observing Tool can be designed to be clever enough to make sure that the minimum necessary calibrations are done to achieve some basic calibration accuracy. Calibration strategies can be recommended or even required by the system based on the required calibration accuracy specified by the user. Each time-contiguous piece of a program must always have a preamble and a postamble to make sure a complete set of calibration data are available.

It is also important that all observers have access to flux calibration information from all programs or that the flux calibration is done by the system in a regular way. Further consideration of this issue will be appropriate once the calibration strategy for ALMA is better defined.

2 The ALMA Simulator

It is important to have powerful tools to aid the novice in mm interferometry. One should be able to specify things like resolution, field of view, and rms in various frequency bands and receive a recommended set of configurations, correlator setups, and integration times. The effects of phase noise and decorrelation under different atmospheric conditions should be included in the

simulator. It would also be important to receive a file with the beam map that is likely to result from the proposed observations. The ideal system would fully simulate ALMA observations of a model source, which could be a Gaussian, or any other model distribution of intensities supplied by the user. This tool would then be extremely useful for data analysis at a later time.

The ALMA simulator will also be invaluable in preparing Phase II proposals. For example, a tool to assist in setting up mosaics and a tool to show the chosen correlator setup overlaid on a simulated spectrum of the source would be very useful. Many tools may be useful for both Phase I and Phase II preparation. Non-standard observing scripts should certainly be allowed for the expert user, although these should be verified as much as possible. At a minimum, the verification process should check that basic requirements such as pre/postambles, calibration, and pipeline processing for the archive are included.

Given the basic policy that ALMA should be friendly to non-expert users, the scheduling blocks generated by the default setting of the software should be good enough that most users, including experts, would be willing to adopt the default setting to generate their scheduling blocks. Recommended settings may be particularly useful for observations of Galactic objects in some of the “standard” lines and continuum, for which the users may just need to specify the pointing center and the required rms noise level.

For complex programs, it would be useful to be able to simulate running a set of interdependent scheduling blocks. This simulation could perhaps function as the validation stage or could be more sophisticated, i.e. a real observing simulator. This test could turn up errors in the specification of the interdependency between blocks, for example. It could be useful to have the simulator include a weather model so that the observer could see a longer program being broken up into several smaller pieces, for example as weather conditions shift from day to day. This type of simulator would check that the preamble and postamble always work properly. An advanced simulator like this could go a long way towards checking non-standard observing scripts.

It is quite clear from the recommendations above that a powerful ALMA simulator is an important aspect of making ALMA maximally useful. It is also clear that electronic data bases must be flexible, yet secure. Much of this technology is available, and we need only adopt it, but the ALMA simulator will be more challenging. Some of the current work on the study of the imaging characteristics should provide a framework for the simulator.

Reccomendation: ALMA should develop a powerful simulator which is capable of a complete end-to-end observing simulation of a project composed of a number of scheduling blocks.

3 The TAC and Review Procedures

At least in the early years of ALMA, it will be important to have a technical review before the scientific review. Use of the ALMA simulator can make much of this technical review automatic, as long as the simulator is updated regularly to take account of ALMA development and experience with actual observations.

3.1 Stringency

Part of this technical review should be a quantitative measure of what has come to be called the “stringency”. The stringency can be defined as t_a/t_p , where t_a is the total observing time available and t_p is the total time during which this project can be done. In practice, t_p will be calculated based on the required water vapor, seeing, pointing, uv coverage, sensitivity, etc. The stringency is then the inverse fraction of the time that the observations can be done, according to statistics that are built up over time. This concept is described in the SSR document. This information should be available to the scientific review panels to aid them in designing a program that has reasonable coverage of the “observing condition phase space”. Filling this space is a well-known problem for observatories operating at submm frequencies. The stringency should be calculable from the parameters given in the Phase I proposal and the ALMA simulator. In

the early operations phase, human judgment may be necessary to apply appropriate corrections, but the goal should be an evolving simulator that does this as automatically as possible. In order to check the Phase II observing scripts versus the approved parts of the proposal, it is essential that the review committee be able to add their recommendations into the electronic data base of the proposal. In this way, it should be possible for the committee to approve only parts of proposals, to assign different priorities to different parts of proposals, etc.

Recommendation: ALMA should adopt the concept of stringency. This concept may be defined as t_a/t_p where t_a is the total observing time available and t_p is the total time during which a given project can be done.

Discussion: Consider further the definition of stringency: have we defined stringency in the best possible manner?

3.2 TAC Operations

From a user point of view, there should be at least two proposal deadlines per year (one per year is very restrictive from the point of view of student thesis work). Individual proposals need to be reviewed both scientifically and technically, and need to be checked for overlap with scheduled proposals and other proposals submitted in the same period. In addition, the proposals need to populate the available observing conditions well and may need to satisfy constraints on the fraction of time awarded to each partner. We suggest that, with a good simulator, the scientific and technical feasibility could be handled by a single reviewer. However, checking for overlap and in particular comparing proposals against available observing conditions is more complex and probably should be an observatory task.

One set of questions concerns the reviewers. Should there be a few reviewers who grade all proposals on scientific and technical merit? Should there be a few groups of observers, with each group reading all the proposals in a single science category? Should there be a large number of reviewers, perhaps 1-3 for each proposal with each reviewer reading 1-3 proposals? Should the reviewers be exclusively ALMA staff, or exclusively not ALMA staff, or some combination? A second set of questions concerns the TAC. Should the TAC be the same people who are also the reviewers (i.e., in the first two cases)? Or should the TAC be a different set of people? These questions probably do not need to be decided now, but the division of tasks between the reviewers and observatory staff will have implications for staffing levels.

The Science Software and Requirements document suggests that “Reviewers should take into account the percentage of observing conditions in each category and accept proposals accordingly.” This is a very large task that will probably be best done by the TAC with help from the observatory. In general, we may need to populate a three-way space (RA, observing conditions, partner share). This task will probably require some clever software to display how things are progressing throughout the semester as well as experienced people to monitor it and potentially tweak the inputs as the semester goes on. One way to deal with this is to reject only truly infeasible proposals in the lowest frequency bands (< 100 GHz, or perhaps even < 230 GHz, depending on the weather statistics at the site). By keeping low priority proposals that can use poor weather, we maximize the likelihood of ALMA always having a source to observe. In fact, we might reject only truly infeasible projects at any frequency and just give a very low rating to poor proposals. One could then rely on the dynamic scheduler itself to ensure a reasonable coverage of observing condition space, as long as enough proposals were available to it.

The TAC will likely need to be able to assign different priorities to different parts of a program. At a minimum, one could envisage an accepted program that happened to include a previously observed source, and so the entire proposal was approved except for a single source. At a higher level, a program might be approved with different rankings for several different sources or for the same source at different wavelengths.

Another question to be considered is whether approved projects are carried over from a previous semester, and if so, should they get a higher priority for completion? This decision might be a complicated function of how close to completion the program is, how high a scientific

ranking it had originally, and what its stringency requirements are. Clearly highly ranked, high stringency, nearly completed programs should be completed before new ones in a similar class are started. It seems reasonable that programs that are highly ranked and nearly completed be carried over and completed the next semester with a high priority, regardless of their stringency.

Discussion: There should be further study of how the ALMA TACs should work, including a review of how existing TACs operate.

4 Flexibility in Phase II and During Observing

One key issue for observers is how to maintain sufficient flexibility to update scheduling blocks, for example, to take advantage of new information gained from other telescopes or to incorporate results from ALMA on the first few sources from a large sample. This desire for observer flexibility will probably be constrained by the need to ensure that the scheduling blocks match the approved proposal and that the set of scheduling blocks is sufficiently stable with time that the dynamic scheduler can operate efficiently. It should be straightforward to make sure that the scheduling blocks match the approved proposal if sufficient information is specified *for each source* in the Phase I proposal. A reasonable compromise between infinite ability to change and a single fixed submission might be the following: scheduling blocks should be changeable until the program is started, and again after any breakpoint is reached.

An alternative approach would be to allow only very limited flexibility in updating scheduling blocks after the proposal has been accepted. The reasoning behind this approach is as follows. Assume that the outputs from reviewing by the Time Allocation Committees (TACs) not only include scientific rating and technical feasibility but also reflect zeroth order fitting of the proposals into expected sets of observing conditions as functions of frequency, LST range, and configuration. The successful proposal may thus be split by the TACs into several parts of different ratings depending on the observing frequency, LST range of the source, and configuration, as well as the scientific merit and technical feasibility of the entire proposal. Allowing too much flexibility to the users after time allocation is complete may reset all these careful considerations and lead to problems with a time-varying ensemble of scheduling blocks. Break points could still be used to allow the observer to evaluate the status of the observations and perhaps update the scheduling blocks. However, these updates would be limited to slight modification of the pointing (i.e. because of possible offsets of the spatial distribution of millimeter/submillimeter sources to their counterparts in other wavelengths) or frequency (e.g., V_{lsr} slightly wrong) or correction of obvious careless mistakes.

One key parameter that observers might not be allowed to change would be the resolution of their observations, since this would affect the configuration schedule.

Discussion: How much flexibility to adjust approved programs should be allowed in the Phase II stage and once observing has started?

5 Setting Priorities in Dynamic Scheduling

The Science Requirements and Use Cases document gives a lengthy list of factors to be considered in dynamic scheduling. To first order, these factors can be divided into two categories: conditions which must be satisfied for the scheduling block to be even considered for scheduling; and conditions which are used to set relative priorities between eligible scheduling blocks. Factors in the first category include things such as LST range (is the source currently visible?) and atmospheric opacity (can the required frequency be observed?). To some degree these types of factors are mostly “go/no go” choices and are fairly straightforward, so we will concentrate here on the second set of factors, those used to set relative priorities.

In setting relative priorities, the most obvious determinant is *scientific ranking*. However, on instruments like ALMA for which certain frequencies are only observable a small fraction of the time, *stringency* should also be an important consideration. For example, projects with lower

scientific ranking that require very good weather conditions may need to be given preference above projects with higher science ranking that can use a wide range of weather (for example, projects at 3 or 7 mm).

A final important consideration is the *execution status* of a project. This factor is designed to give some priority to completing projects which require only a small amount of time to be finished. This is a particularly important consideration for the project that is currently being executed. For example, suppose the weather suddenly changes from 3 mm weather to 350 micron weather. How quickly the scheduler should decide to stop the current project and move to one that will take advantage of the better weather should be a function of the status of the current project. For example, suppose the current project would be completed in just 10 minutes (perhaps 5 minutes of source observations and 5 minutes of postamble observations). It would probably make sense to complete the current project, rather than stopping (and still needing to spend 5 minutes on the postamble), and then spending perhaps 15 minutes at a later time (now including preamble observations as well). On the other hand, if the current project still needed 30 minutes to finish, it would probably be most efficient to wind the current session up quickly and move along. It is clear from this example that the typical time needed to be spent on preamble and postamble observations associated with each session on a project will influence the exact timing of these decisions. Also, completion in this context may usefully be defined as completion of the project as a whole, completion of a single source in a larger project, or completion of a single source in the current configuration of a multi-configuration project.

We suggest the following three factors are the important ones to be considered in designing the dynamic scheduler.

1. scientific ranking
2. stringency
3. execution status

These should not be considered absolute, but should be assigned some weights. These weights might be different over different timescales. For example, execution status might be weighted most highly if the current program required only 5 minutes to finish, while stringency and scientific ranking are clearly more important factors on timescales of days or weeks, respectively.

Another consideration is whether to include a delay in the situation when the weather changes. For example, if the weather is slowly degrading, should we continue to observe the current project for some short period of time in marginal weather before switching to a project that is better suited to the current weather? Similarly, if the weather seems to be improving, how long to we wait to make sure it is going to continue improving before switching to a new program? Again, both these decisions will be affected by the overhead involved in switching programs too often, i.e. the time spent in pre and postamble observations.

Finally, unlike what is described on page 6-8 of the ESO Operations Proposal, it seems likely that eventually the dynamic scheduler will have to be almost completely automatic (i.e. *not* prioritized in real time by a support astronomer). This type of real-time prioritization by a support astronomer is hard enough to do at existing telescopes like OVRO and the JCMT which have programs using blocks of several hours and only a range of 3-4 in frequency; it seems likely to be impossible for ALMA. However, in the early days of ALMA, when we are still learning about the weather conditions and the algorithms, significant human interaction with the dynamic scheduler will likely be required.

Recommendation: The dynamic scheduler should include science ranking, stringency, and execution status as three of its key parameters.

Discussion: How should the three key priorities for the dynamic scheduler be balanced? Are different weightings required on different time horizons? What policies do existing observatories follow?

5.1 Dynamic Scheduling of the ACA

The ACA will likely also require dynamical scheduling, and so this needs to be considered in the studies as well. Some of the parameters of the ACA will likely be somewhat different from the main array. For example, the ACA will likely observe fewer projects but for longer periods of time. Depending on how long is required for each project, this could make it harder for the dynamic scheduler, for example, if the demand for very good weather became very high.

If the ACA and the main array are run by two independent dynamical schedulers, there will likely need to be some passing of information between the two. For example, a program which has completed its observations with the main array might get a higher priority to complete its observations with the ACA and vice versa.

Recommendation: A dynamic scheduler for the ACA needs to be included in the software planning.

6 Support while Observing

ALMA observing will be “Service Observing” for a variety of reasons (primarily logistic). One of the penalties which one pays for service observing is that one cannot react in the same fashion to the “unexpected” where the unexpected may be the astronomical results (we do not know ahead of the observations what we are going to find) or technical or weather related problems. One question which ALMA operations will pose is whether the inevitable loss of efficiency which service observing involves can be minimised (efficiency in this sense implies efficiency in reacting to the astronomical results). This needs to be done in a manner which does not cause the efficiency of ALMA operations in general to be reduced.

To a large extent, ALMA should base its policy on experience in existing institutions (Plateau de Bure, BIMA, OVRO, NMA). It is true that ALMA will be “different” but most of these differences will only emerge after actual experience has been gained. The SSR report indeed makes use of that experience and seems a good zero-order attempt to outline a reasonable approach which ALMA might adopt. The following are merely some reflections on the ways in which ALMA might interact with observers during actual observations.

One can envisage two ways in which observers can be involved in real time in ALMA observations.

1. Look at the data using standard (pipeline) reduction procedures and communicate to the ALMA operations center if things look wrong. We will call this “eavesdropping” and some options for implementing it are discussed in more detail below.
2. Pre-program break points in the observations at which one would stop taking data for a certain minimum period (say 24 hours) thus allowing a more balanced look at the data quality and results.

The first of these is clearly beneficial in that experience suggests that *only* the observer is in practise sufficiently motivated to note unexpected features in the data. It involves some organisation in that with dynamic scheduling, one will never be quite sure when a certain set of observations will be carried out. However, it should not require data handling by the operations center over and above that needed by the staff checking data quality. It requires a standard pipeline package that can handle in real time programs with a reasonable data rate. This should allow the observer to decide whether his (her) scientific goals are being reached and to communicate rapidly with the operations center in the event that something is going wrong.

The question of break points is trickier and in our opinion must be handled in a manner which allows the (human) scheduler in the ALMA control center the final decision. Observers cannot be allowed to break off every 5 minutes to look at the data! More importantly, ALMA operations needs one person in charge who decides on a day-to-day basis what happens. On the other hand, one can envisage cases where for example, a follow-up observation should only be made in the case that a certain source is detected. The observer could program the observations

in such a way that there is a break-point when the critical RMS for detection has been reached. He(She) must communicate the result to the operations center within a certain time subsequent to the break-point. Breakpoints will likely be *required* for long programs as well as in the early years of ALMA, to protect against wasting large amounts of observing time.

6.1 Different possible levels of Eavesdropping

1. At a minimum, eavesdropping means being notified that your project is now being observed, and monitoring the images as they come out of the pipeline in real-time, perhaps by looking at a website. This option is the minimum required and is currently mentioned in the Software Requirements and Use Cases document.
2. A second level of complexity would be to allow the observer to phone the operator to say that something is going wrong and the observation should be stopped until the observer can figure out what it is. This could function within the current scheme by the operator being able to insert an instantaneous breakpoint into the program, and the program will not be scheduled again until the observer clears the breakpoint. How the operator inserts this breakpoint would need to be worked out; presumably the observer could then clear it in whatever way is used to clear pre-planned breakpoints. This option might be a good one to have, but the SSR people would need to figure out how it could be done.
3. A third level of complexity would be to allow the observer to make real-time decisions that are something OTHER than pausing the observations until a later date. An easy example would be when the observer is measuring a long list of objects, and one of them comes in brighter than expected. The observer might want to stop observing that source (i.e. the observations are deemed complete for that source at that frequency) and move on to the next source in his/her list. Alternatively, something might not be detected that was expected to be, and the observer might want to integrate longer at the expense of doing fewer targets. Some of this might be handled by the use of preset breakpoints if the system is clever enough; for example, in the first case, you could specify $rms \leq x$ OR $T(int) \geq y$ OR $S/N \geq z$, where any one condition causes the observations to cease. It is harder to see how you would specify the second case, where one needs to integrate longer ($S/N < z?$), but it might be possible. The observer would also need to be clever about how he/she uses the software. If these decisions could be handled by clever use of breakpoints, it may not require any software and people beyond what would be required for the interrupt option described above. We need to examine how clever we can make the breakpoint software while still maintaining a robust system. A sophisticated simulator would provide a vital check of complex interdependencies between scheduling blocks.

Discussion: How should breakpoints and/or eavesdropping be implemented from a user's point of view?

7 Regional Data Centers (RDCs)

After observations are obtained with ALMA, the astronomer will need continued interaction with the "observatory". This interaction will be of varying degrees, depending on the experience of the astronomer and the type of project undertaken. This interaction is primarily related to obtaining the data, whether pipeline reduced images or raw visibilities, along with any ancillary data ("archiving") and for use of or assistance with data reduction and/or analysis ("data analysis support").

The facilities the astronomer will utilize in this stage include one or more of the three Regional Data Centers (RDC), nominally located in Europe, Japan and North America, along with the astronomer's personal workstations or home institute's other computing resources. A working-model of the RDC is given in the ESO operations proposal. We wish to provide advice in more detail on the possible types of interactions that could arise and should be supported.

The major roles of the regional data centers are as follows:

1. The Data Portal – The first and main function of the RDCs are to facilitate the transfer of data from the Array to the User. The current plan (see the ESO operations proposal) is that each RDC receives all the observed data from the Science Operations Center (SOC) and creates a “mirror” archive, while the SOC keeps the master archive. These archives include cleaned images as well as the raw and calibration visibility data and other array data (weather, etc), e.g., as requested in the Phase 2 Proposal Process (P2PP) obtained under the proposal of the astronomer. A second option to be considered is for the RDC to mainly supply a gateway to the archive without keeping the actual data (for example, compiling an archive of only the header data for use in search and location of specific data). In this case only the SOC would hold a true archive, and the load in hardware at the RDC could be considerably less. In either case, the access to the RDC for data retrieval is basically to be made remotely by the astronomer.
2. Software Development – One goal of the RDCs will be to work on improved algorithms for data reduction, analysis (through Aips++ or other package) and archive mining, as well as development of tools for interaction with existing or future archives, such as the National Virtual Observatory in the US (see §3.2). This includes not only an interface for retrieval of data from these archives, but a facility for transfer of basic pipeline reduced data into the larger, multi-band (NVO-type) archive.

The software should have the goal of running not only at the RDCs but also be portable to platforms at the workstations of home institutions. The data rate we can handle between the RDC and the home institute should increase quite a lot by 2008-2010. However, in the early stages of ALMA it may require more reduction and analysis “over the net”, particularly for researchers at smaller institutions.

3. Analysis Support – The RDC will provide help remotely to the users. The help should span the range of simple advice related to the default pipeline data reduction algorithms, as well as more sophisticated requests, such as using advanced or specific algorithms for rereduction of the data. An issue here is whether the centers should supply computing resources for really big reductions over the net. These interactions should be basically fulfilled remotely, but those who hope to get deeper into the processing may want to come and stay at the RDC for a while. The RDC should then be able to support them on a face-to-face basis.

The relative roles of the individual RDCs has yet to be formally defined. ALMA will need to balance the need to provide efficient (and perhaps “local”, meaning within the continent) services to astronomers against the cost of supporting three RDCs. The former is possibly best handled by having the RDCs remain quite similar (the “Clone” model) with nearly identical resources, capabilities, etc. In the face of limited resources, it may make for more cost-efficiency for each of the RDCs to have areas of specialization, with some necessary “core” capability at each of the three (the “Distributed” model).

Each has pitfalls. For example, the Clone model suggests a high degree of redundancy that may not be necessary, as well as the need for some sort of control to maintain the uniformity of the centers. The Distributed model may force an astronomer to interact with an RDC many time zones away, which may be inconvenient, and in some cases may require the astronomer to travel to the RDC of a different partner for face-to-face assistance. A more significant danger of the distributed model is that the capabilities and compatibilities of the RDCs will have a tendency to diverge, and a strong control will be needed to ensure that they don’t wander too far from each other.

At this time, we recommend a compromise of sorts, suggesting that the centers should strive to have a high overlap of capability, and thus lean toward the Clone model. We note that it is natural for the support astronomers of each center to have their own areas of expertise, and that users may seek out the “world-expert” in some area of reduction, analysis, or computing, no matter where they reside.

A final, rather subversive option has been proposed regarding the RDCs, in that the scientific need for three (one for each partner) is not as yet well-substantiated. We can conceive of integrating the three RDCs into a single Data Center, located within the scope of any of the three partners, or even at or adjacent to the SOC in Santiago. From the standpoint of the human (political and social) elements, the need for three is justifiable, as the long-term RDC staff from each of the three partners would presumably prefer to live closer to “home”. The RDCs also represent the most visible structural elements of ALMA for the general public of each of the partners.

Recommendation: Three Regional Data Centers (RDC), are to be located in Europe, Japan and North America and are responsible for data portal, software development and analysis support. They should be operated with an international and collaborative spirit.

Discussion: The relative role of the RDCs: The extreme of the RDCs may be either of the “Clone” type or of the “Distributed” type. At this time we recommend a compromise of sorts, suggesting that the centers should strive to have a high overlap of capability, and thus lean toward the Clone model. This needs more discussion.

Discussion: Should the RDC keep all the actual data or supply a gateway to the archive without keeping the actual data?

Discussion: Should the RDC supply computing resources for really big reductions over the net or not?

Discussion: Do we really need 3 RDCs?

8 Archives

The role of the ALMA archive should be twofold. One is for the pre-observing users to learn what has already been done and what is planned to be done in a given observing session. The other is the real archive of all the ALMA data open to the worldwide community anytime.

The real ALMA archive may be further divided into two parts. The first includes the images produced by standard scripts (see also Sect.1). The second includes the images produced by the observers, which may contain substantially enhanced images or other relevant data products. The responsibility for the second part should belong to the individual observers. It needs further consideration if the second part should remain within the official ALMA framework or rather should be organized outside it.

The ALMA archive should be fully compatible with the Virtual Observatory (VO) and the Grid paradigm on which the VO is based. ALMA will be the first major observatory coming on-line post-VO. In the VO context, the ALMA archive data should be available independent of location and there should be no distinction between the “master” and “satellite” archives. Through the Grid, the astronomer’s desk-top computing power can be enhanced relative to what he/she has available locally. If ALMA is going to provide processed data in a user-friendly way to a non-expert community, then it should take advantage of the VO environment and the underlying GRID technologies.

There are two major and distinct development areas:

1. development of the post processing, quality control and data analysis software;
2. development of the archive for the post-VO and Grid era.

Operationally, one may distinguish between the following entities:

1. A Science Operations Center (SOC), responsible for post processing, quality control, and delivery of the data products to the astronomer and archive. It is an **operational** center, and as such is not necessarily involved in software development. It is essential that there be only one such center, to assure the homogeneity and quality of the final data products. Its location is a secondary issue. Chile seems a logical choice, as it is “neutral ground”

for the project (in which case Santiago would be preferred, as it is easier to recruit staff there), but this is not essential.

2. Three Regional Data Centers (RDCs), in Europe, Japan and North America. These RDCs could share software development, and could provide assistance to users in their own region (Europe, Japan, North America). In the VO/Grid era astronomers should have easy access to the archive wherever they are (that is just a matter of bandwidth), but they may also require assistance from the experts in their RDC, and in some cases may travel to the RDC for hands-on assistance (see 7).

Recommendation: The ALMA archive should be open to the worldwide community and be fully compatible with the Virtual Observatory (VO) and the Grid paradigm.

Recommendation: The Science Operations Center (SOC), should be responsible for post processing, quality control, and delivery of the data products to the astronomer and archive. It may not be necessarily involved in software development.

Discussion: Should images produced by observers, as well as those produced by standard scripts, be placed in the official ALMA archive?

Discussion: The location of the SOC: Chile seems a logical choice, as it is “neutral ground” for the project (in which case Santiago would be preferred, as it is easier to recruit staff there).

9 Proprietary period

We recognize that a fairly short proprietary period will help ALMA to have an early impact, along with the production of quality pipeline images (as opposed to quality pipeline visibility data), allowing astronomers of all flavors to utilize ALMA with minimal discomfort and maximum scientific weight. A proprietary period of regular projects should be 1 year as is commonly used in the currently working instruments, with some exceptions.

A key question here is when the clock starts to count. The simplest solution is to use the time when all the observations in a project are completed on ALMA. This perhaps works for most of the projects of short observing times. We need to consider the effect of proprietary periods on long-term projects extending over a long time frame (years), including Key or Legacy projects. The term legacy might be taken to mean large blocks of time in exchange for no or very short proprietary period. It should be avoided as much as possible that the community cannot see the data for years from such long-term projects, since they are often valuable and of strong impact on science. We therefore should consider setting an upper limit for the proprietary period like 2 years from the first day of observation for any long-term projects.

We do need to impose a longer period for thesis projects. This may need different consideration, and we suggest 3 years as the proprietary period since the start of the observations for thesis projects.

Recommendation: A proprietary period for regular projects should be 1 year as is commonly used in the currently working instruments, with some exceptions for legacy projects and for thesis projects.

Discussion: We need to consider the effect of proprietary periods on long-term projects extending over a long time frame (years), including Key or Legacy projects.

Discussion: Proprietary periods for thesis projects; We tentatively suggest 3 years as the proprietary period since the start of the observations for thesis projects. This needs further discussion including cases like a thesis project as part of a legacy project.

10 Recommendations

1. Complete information on the source parameters (coordinates, velocity, frequency, resolution, rms noise) in approved and completed projects (both proprietary and public) should be available in the archive.
2. ALMA should develop a powerful simulator which is capable of a complete end-to-end observing simulation of a project composed of a number of scheduling blocks.
3. ALMA should adopt the concept of stringency. This concept may be defined as t_a/t_p where t_a is the total observing time available and t_p is the total time during which a given project can be done.
4. The dynamic scheduler should include science ranking, stringency, and execution status as three of its key parameters.
5. A dynamic scheduler for the ACA needs to be included in the software planning.
6. Three Regional Data Centers (RDC), are to be located in Europe, Japan and North America and are responsible for data portal, software development and analysis support. They should be operated with an international and collaborative spirit.
7. The ALMA archive should be open to the worldwide community and be fully compatible with the Virtual Observatory (VO) and the Grid paradigm.
8. The Science Operations Center (SOC), should be responsible for post processing, quality control, and delivery of the data products to the astronomer and archive. It may not be necessarily involved in software development.
9. A proprietary period for regular projects should be 1 year as is commonly used in the currently working instruments, with some exceptions for legacy projects and for thesis projects.

11 Topics for discussion and further study

1. Consider further the definition of stringency: have we defined stringency in the best possible manner?
2. There should be further study of how the ALMA TACs should work, including a review of how existing TACs operate.
3. How much flexibility to adjust approved programs should be allowed in the Phase II stage and once observing has started?
4. How should the three key priorities for the dynamic scheduler be balanced?
5. How should breakpoints and/or eavesdropping be implemented from a user's point of view?
6. The relative role of the RDCs: The extreme of the RDCs may be either of the "Clone" type or of the "Distributed" type. At this time we recommend a compromise of sorts, suggesting that the centers should strive to have a high overlap of capability, and thus lean toward the Clone model. This needs more discussion.
7. Should the RDC keep all the actual data or supply a gateway to the archive without keeping the actual data?
8. Should the RDC supply computing resources for really big reductions over the net or not?
9. Do we really need 3 RDCs?
10. Should images produced by observers, as well as those produced by standard scripts, be placed in the official ALMA archive?
11. The location of the SOC: Chile seems a logical choice, as it is "neutral ground" for the project (in which case Santiago would be preferred, as it is easier to recruit staff there).

12. We need to consider the effect of proprietary periods on long-term projects extending over a long time frame (years), including Key or Legacy projects.
13. Proprietary periods for thesis projects; We tentatively suggest 3 years as the proprietary period since the start of the observations for thesis projects. This needs further discussion including cases like a thesis project as part of a legacy project.