Integrating a university team in the ALMA software development process: A successful model for distributed collaborations

Matias Mora[a,b], Jorge Ibsen[c], Gianluca Chiozzi[d], Nicolás Troncoso[a,b], Rodrigo Tobar[d], Mauricio Araya[e], Jorge Avarias[f], Arturo Hoffstadt[b]

[a]Associated Universities, Inc. (AUI), Santiago, Chile;
[b]Universidad Técnica Federico Santa María (UTFSM), Valparaíso, Chile;
[c]European Southern Observatory (ESO), Santiago, Chile;
[d]European Southern Observatory (ESO), Garching bei München, Germany;
[e]National Institute for Research in Computer Science and Control (INRIA), Nancy, France;
[f]National Radio Astronomy Observatory (NRAO), Socorro, NM, U.S.A.

ABSTRACT

Observatories are not all about exciting new technologies and scientific progress. Some time has to be dedicated to the future engineers' generations who are going to be on the front line in a few years from now. Over the past six years, ALMA Computing has been helping to build up and collaborating with a well-organized engineering students' group at Universidad Técnica Federico Santa María in Chile. The Computer Systems Research Group (CSRG) currently has wide collaborations with national and international organizations, mainly in the astronomical observations field. The overall coordination and technical work is done primarily by students, working side-by-side with professional engineers. This implies not only using high engineering standards, but also advanced organization techniques.

This paper aims to present the way this collaboration has built up an own identity, independently of individuals, starting from its origins: summer internships at international observatories, the open-source community, and the short and busy student's life. The organizational model and collaboration approaches are presented, which have been evolving along with the years and the growth of the group. This model is being adopted by other university groups, and is also catching the attention of other areas inside the ALMA project, as it has produced an interesting training process for astronomical facilities. Many lessons have been learned by all participants in this initiative. The results that have been achieved at this point include a large number of projects, funds sources, publications, collaboration agreements, and a growing history of new engineers, educated under this model.

Keywords: Software engineering; Geographically distributed, collaborative software development; Project management; Organizational model; Student organizations

1. INTRODUCTION

Astronomical observations require normally particular atmospheric conditions, which are found in few places on earth. One of the most popular countries for astronomy today is Chile, specifically its Northern regions. Large astronomical facilities have caught the attention of the international community in the last decades. As international research facilities, astronomical observatories have become attractive for Chilean Engineers, specially motivated by this scientific work environment. While the local astronomy community has been growing significantly over the past years, Engineering is still lacking some sort of specialization in Chilean universities, although the observatories' on-site Engineering staffs are today mostly Chilean.

Further author information: (Send correspondence to Matias Mora)
Matias Mora: E-mail: mmora@alma.cl, Telephone: 56 2 467 6538

Like other areas, major observatories feature summer jobs for university students. This is actually a quite good way to identify high quality and motivated future professionals. The Software Group at La Silla, part of ESO facility La Silla - Paranal Observatory (LPO), was following this same practice in the early 2000's, but with the motivation of doing something more than just isolated summer jobs, with little or none impact on the project. After some failed attempts of motivating students to think about long term projects, related to their Universities life, finally some students from Universidad Técnica Fedrico Santa María (UTFSM) took the opportunity and started building up a group in Valparaíso. The initial motivation was to take advantage of a new project, in which ESO was also involved, that would be starting soon its construction in Chile: the Atacama Large Millimeter/submillimeter Array. ALMA, an international radioastronomy facility, is a partnership of Europe, North America and East Asia. Construction and operations are led on behalf of Europe by ESO, on behalf of North America by NRAO, and on behalf of East Asia by NAOJ. The Joint ALMA Observatory (JAO) provides the unified leadership and management of the construction, commissioning and operation of ALMA.

Through internal contacts with the Software Group at La Silla, a small collaboration with the ALMA Computing group in Garching, Germany was initiated. This was mainly possible because of ALMA's decision to make its distributed control framework, ALMA Common Software (ACS),[1] an open-source product. This general purpose platform is also used by other physics and astronomy projects around the world. On the opposite, LPO was using the VLT (Very Large Telescope) Software, which could not be easily shared with an interested community. This was the beginning of the "ACS-UTFSM Team", in 2004, driven by curiosity to explore software frameworks used in large astronomical projects, mainly focused on distributed systems. The initial motto was "Just for fun!", as unpaid spare time activity. The first main project of this 3 to 5 people team was to explore the possibilities of a "Generic Telescope Control System" (gTCS), based on ACS, which had been thought as a reusable control system, based on ACS. At the same time, some nice-to-have applications were delivered to the ACS project, and ACS-UTFSM was recognized as one of its official development teams, distributed through all over the world.

After three years of informal work, the team presented a technology exchange initiative to the ALMA-CONICYT Fund 2006, including development of ACS tools and gTCS related research, which was granted in 2007. This fund aims to support development of astronomy-related projects at Chilean universities. Through this financial support, the group got also recognized by the Informatics Department, providing it with a dedicated laboratory. That same year, the now called "ALMA-UTFSM Group" became part of the new Computer Systems Research Group (CSRG),[2] an initiative founded by the same group of students, but with broader objectives. The group grew, and involved about 20 students, making also its organization more and more complex. In 2007, the ACS Workshop, a training and discussion instance for the ACS community (including various observatories and other universities), was first time organized outside of the ESO headquarters in Germany: it went to Valparaiso (see[3]). With this event, the group acquired international visibility. During the next years, other funds were granted, and the group consolidated itself as a collaboration partner, also involving other organizations like NRAO, and as a strong students initiative inside UTFSM. More recently, ALMA-UTFSM is working on some new development areas, like code generation, and research areas mainly related to applied Artificial Intelligence.

Until now, more than 40 students and 5 professors have participated in the group's activities over time. Also, 7 former CSRG members joined the ALMA Computing Group in Chile and other development sites in Europe and North America, while another 2 are working for ALMA at UTFSM in Valparaiso, paid by NRAO and ESO through agreements with the University. At this point we should point out the difference between CSRG and its now sub-associated ALMA-UTFSM Group. As CSRG has become a wider project development platform, it has expanded itself to other areas than observatory-related engineering. We will therefore center this paper on the ALMA-UTFSM Group (see[4]) activities, although referring to it sometimes as CSRG in general terms.

What complexity does this ambitious project involve? First of all, ALMA and CSRG are meant to work in a distributed manner, not only geographically, but also temporally. This obligates both organizations to make use of proper technological tools. Another challenge is the organization of a group that is mainly managed by students. While most of the working style applied by CSRG belongs to the Software Group at La Silla, and recently the JAO Computing Group, many lessons for both organizations have been learned in these years. Section 2 describes the organizational culture, complemented with the organization model in section 3, and the technological interaction strategies in section 4. Finally, some successful experiences and the expansion of the model are discussed in section 5.

2. ORGANIZATIONAL CULTURE

While each individual has his own personality, an organization has its own culture, which describes its psychology, attitudes, experiences, beliefs and values. By identifying the cultural characteristics of a consolidated organization, we can learn about common objectives and behaviors of its members.[5] The CSRG was built upon existing models and each of them has more or less deeply influenced its current culture.

First of all, ESO and specifically the Software Group at La Silla have been a major influence since the first internship experiences. The CSRG working style was mostly directly adapted from La Silla. This included technical standards, as well as attitudes and values. In particular, staff at La Silla had a strong interest in expanding their frontiers, applying their knowledge and skills to creative personal projects, and avoid sticking only with their routine support work. Summer jobs organized at the La Silla site had a major impact on the motivation of participating students, derived from the highly uncommon and exciting work environment at an international scientific facility. Besides the technical interest, also personal relations are very important at observatories in general, as people are mostly working in shifts and have to spend many more hours with their colleagues than at any normal office. Therefore, personal skills are critical when considering new applicants. Another aspect that has been taken over from ESO is the naming for projects (specially with amusing abbreviations) and roles definitions. Also, all official communications are done in English, as the universal language across the interacting parts.

Derived from the Software Group at La Silla, the ALMA JAO Computing Group (see[6]) has been deeply influenced by the first one, as about half of the members moved from ESO, and more recently from CSRG (see section 5.6). Also, as LPO has reduced its on-site staff at the La Silla site during the past years to a minimum (actually, software support is today done remotely from the VLT site), naturally the main interaction of CSRG is today with the JAO Computing Group, keeping also most of the traditional working style alive. Since 2009, ALMA is offering students summer jobs at its Operations Support Facility, in northern Chile. These jobs started with some UTFSM students working with the JAO Computing Group, and have been recently expanded to other Engineering areas and universities.

On the UTFSM side, all of the first-generation CSRG members started working together way before the group's foundation. UTFSM Informatics Department has a long tradition in open-source technologies, used for the main computational services provided by the department to its members (professors, students and officers). Central services are provided in a Linux based environment (servers and user workstations), and administration is almost entirely done by students, who can earn a huge experience in system administration and open-source development before actually graduating. This implies a close relation to the open-source community, which is known for its distributed work style and high-quality expectations. Naturally, the openness for whatever high-quality contributions from whomever they may come, is intrinsic to open-source projects, and the group as well. Like the later work in CSRG, these jobs implied many more work hours than actually paid, mostly because of the enthusiasm for new experience. This was also the starting point for the tradition of summer jobs at the La Silla site, starting in 2002, as successful students recommended high-qualified colleagues from these university activities for the next year, and so on, paving the way for future opportunity windows. Ultimately, internships as part of the Software Group at La Silla, and the very good synergy and technical culture inside these existing groups, convoluted to the creation of CSRG.

In general terms, CSRG is an important part of the education of its members, and has also a visible prestige within the university, as an elite development and research group. Common values across all CSRG members are commitment, initiative, persistence and courage, while a certain degree of meritocracy is applied. It is important to notice that not only technical know-how is important to stand out, but also personal skills are essential. Feedback about the work performance is given in formal instances, like periodical meetings, but also in informal situations during work interaction. Usually, members who do a good job do not get open recognition of achievements, but get better considered when taken into account for higher responsibilities. This implies a natural selection inside the group, where lower qualified members will eventually have to leave, if they are not able to maintain the group's standards.

Further, the university student's life has some properties that make any organization based on students very different from others. An engineering degree at UTFSM is about 6 years, plus 2 years of optional master's degree. Considering that a student enters the group in average during his fourth university year, we can conclude a limited permanency inside the group. Therefore, internal teamwork and sharing of everything that is learned by the group members among the rest of the team, is essential to maintain a coherence in time. Also, the timing is clearly very different from what we can expect from industrial working environments. First, Chilean universities have a 3-months summer vacations period, which means an interruption of continuous development processes. And of course, there are the exams periods, which will also lower the attention that can be paid to extra curricular activities. Nevertheless, a good communication, regular work-logs and the use of dynamic technological platforms have permitted to overcome this.

These characteristics identify CSRG as an organization, and have already survived several generations of students. To make sure that these characteristics are preserved, new members pass through a special selection methodology to demonstrate their genuine commitment to the project. The internal selection process of new participants involves the whole team, preserving group identity independently of individuals. Every year, several new members are incorporated, but their initial duties are usually tangent projects to the mainstream research and development. These projects are not funded neither paid, but the same quality standards are demanded during this training period. In a maximum 6-months period a natural selection occurs, and some of the candidates leave because of lack of interest, time or technical and interpersonal skills. This way, the risk of loosing quality on the official projects is mitigated, and the results of the tangent projects are usually new research, development and funding opportunities. The remaining newcomers, after this training period, are finally incorporated to official projects, assuming responsibilities in the internal and external communication, coordination and technical duties. The historical reason for this qualification period is the first period of the group's formation, when its original members worked the first three years without any funds nor payment, with the only motivation of eventual opportunities and the excitement of collaborating with international engineering projects.

3. ORGANIZATIONAL MODEL

The organizational model of the Computer Systems Research Group has evolved from a 3-persons group with one coordinator, to a complex structure of about 20 students from different engineering areas and a broad variety of projects. Starting from the cultural factors described in the previous section, we can identify as key element on one side a self-organized group, mimicking the organization of a small development team, and on the other side, a strong commitment of contact persons inside the collaborating organization (ESO/ALMA). The role of the latter is to facilitate the interaction with professional engineers and the development of collaborative projects, as well as participating in high-level decisions of the group's management. Although the collaboration started on a strictly technical level between both parts, at certain point higher level authorities at both sides became involved. This is very important to open more opportunities and carry out more ambitious initiatives.

The overall development inside the ALMA-UTFSM Group is done in integrated project teams, reporting to the management. Each one has a project head, and the project participants are mainly assigned according to personal interests. As of today, we can identify a horizontally organized group head, including four elementary roles (see Figure 1):

- **Public Relations Lead:** In charge of all external interactions and collaborations, acting as group representative. This includes maintaining an active communication with the collaborating organizations, as well as exploring new opportunities and funds. Also in charge of spreading information about the group's activities with public impact. As of historical reasons, the Public Relations Lead has taken over most of the group's strategic planning (project management).
- **Technical Lead:** Responsible for the internal group coordination, in terms of organizing and supervising all active development projects. Has to maintain a constant communication with all development teams, and maintain the status information updated. Also, leads new projects formulations, and has to collaborate closely with the Public Relations Lead in tasks like new funds applications.

- **Management Officer:** In charge of most of the routine administrative activities, supporting the Technical and Public Relations leads. This may include the preparation and reminder of meetings, supervision of action points, and general funds administration. Also, a main contact point with the university or faculty internal administrative officers.
- **Document Officer:** Responsible for the group's research activities, including exploration of new areas with growing relevance. This includes publications in conferences, journals, and any kind of scientific show-window. The Document Officer will facilitate in this context the elaboration of such publications for any member and development team.

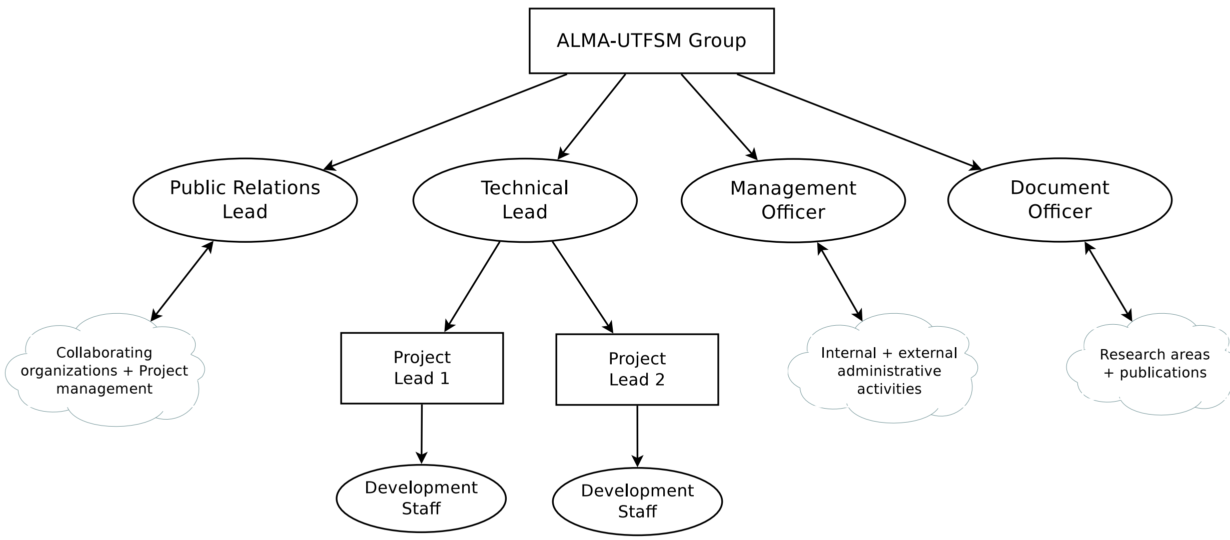

Figure 1. ALMA-UTFSM Group organization chart, as of 2010.

All leading roles have been always taken over by undergraduate students, typically within the last 2 years of their degree. But of course, there has to be a counterpart within the faculty. The faculty (in this case, mainly the Informatics Department) supports the group's activities through the involvement of professors and insertion of relevant activities in the study program, including courses and internships. Also, external funds have to be normally applied with a full-time professor as project head. In this sense, professors have had a relevant role in terms of technical guidance for individual projects (specially in research areas), but also as active support for many university internal activities. The Informatics Department also provides the physical space, a dedicated laboratory, for the group's work at the University, and since 2009, it has included the basic operations costs of CSRG as part of its annual budget.

Moreover, in 2008 CSRG was one of the founding members of the "Academic Students Initiatives Program" (PIE>A, as of its Spanish acronym).[7] This program gathers some of the major UTFSM students initiatives, and represents their interests as a broader university organization. During the last years, PIE>A has grown to a relevant part of the educational project of the University, being the most active "learn-and-do" actor. This includes other groups, like the "Robotics Center" (CR) or the "Alternative Energy Generation" group (GEA). The program is today part of the UTFSM Teaching Direction, under the supervision of the Academic Vice-Rector. As part of PIE>A, CSRG does not necessarily depend on the faculty authorities, and allows a university-wide action, and also the necessary importance to manage high-level collaboration agreements.

It is important to notice that this recognition from the faculty and university side was not easy to achieve at the beginning, as an internal organization led by students was something unusual, and there was no confidence that these ambitious young people could develop important projects and collaborations with other organizations. Although history has shown the opposite, we can probably expect a similar difficulty at other educational institutions.

4. TECHNOLOGICAL MODEL

How does this organizational model finally interact with its collaborators at the technical level? We can identify mainly four different collaboration approaches that have been used during the history of the group, normally found as combinations of themselves. These distinguish mainly differences among people (developers) and resources (development environments, servers, electronic circuits, etc.) distribution, which implies both geographical and temporal (time zones) distances:

1. **Distributed communication and centralized endpoint:** In general terms, development is done in several geographical locations and by several people at the same time. To facilitate this distributed interaction (even if in one same location), normally some sort of centralized platform is used. This permits to make all relevant information easily available at this centralized endpoint to all team members, without passing through an explicit centralized authorization. Information typically shared through this mechanism include general project information and planning, regular work-logs and discussions (see figure 2).

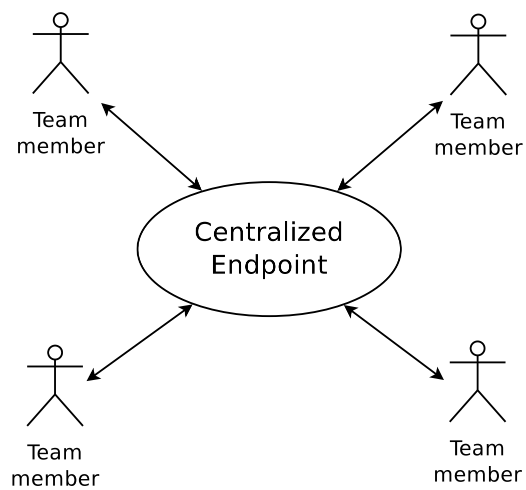

Figure 2. Distributed communication and centralized endpoint.

2. **Geo-temporal distributed individual work:** Sometimes a work can be partially performed individually, but with resources located both locally or remotely. This applies for example to specific projects, where the immediate workspace of the developer has not all the needed resources. In this case, resources available at other places can be shared remotely over dedicated networks, or even the Internet (see figure 3).

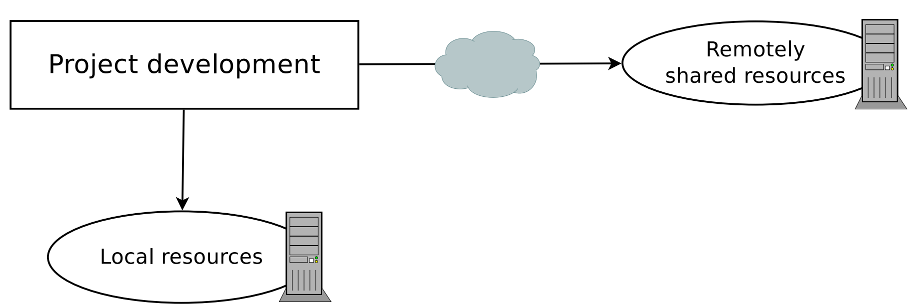

Figure 3. Geo-temporal distributed individual work.

3. **Geo-temporal distributed teamwork:** Usually, the teamwork will occur in several geographical locations at the same time, but a distributed team can be involved in particular projects that have an equal amount of collaborative work on each side. We can distinguish between a supervised and a non-supervised approach. The first one implies that one of the participating parts will supervise the work of the other parts, while the latter will be based on a self-supervised work, where all parts are in charge of simultaneous development (see figure 4).

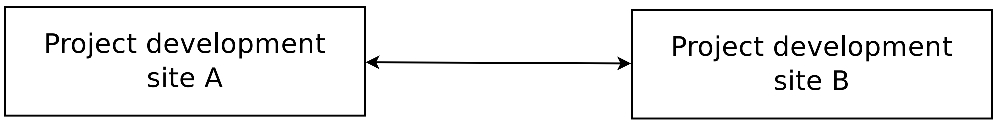

Figure 4. Geo-temporal distributed teamwork.

4. **Non-distributed teamwork:** The simplest approach is to have one single team working on an isolated project in one single geographic location. Obviously, this is also the less common case. This may apply mostly to new initiatives, that are not yet mature enough to be shared with other teams or organizations (see figure 5).

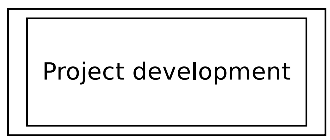

Figure 5. Non-distributed teamwork.

As can be seen, the interaction is obviously most of the time distributed, or implies at least some distributed resources. It is therefore necessary to use adequate technological tools to make this interaction possible and facilitate efficient work. The main technologies constantly in use by both sides can be summarized as follows:

- **Operating systems:** The most used system across workstations and servers is GNU/Linux, usually in its RedHat (Fedora, CentOS), Ubuntu and Debian flavors. Also, some recent development has needed the MS Windows platform. Finally, real-time research has been done using VxWorks , QNX Neutrino, RT-Linux and RTAI.
- **Communication tools:** An efficient communication is essential for distributed work, and includes instant communication, as well as collaborative platforms. Among the traditional technologies, there are, phone, e-mail and mailing lists. Taking advantage of Internet resources, the most used instant communication platforms include Yahoo! Messenger, GoogleTalk and Skype, all of them freely available and multi-platform. Finally, on-line collaborative platforms include TWiki and Google Calendar.
- **Development tools:** Distributed or individual development both need an efficient platform and certain sharing mechanisms. Version control systems are the most essential part of this, and have included intensive work with CVS, SVN and git. More recently, project management systems like Trac have been explored as an integrated solution. Although first architecture designs are always put on the drawing board (or scanned for on-line discussions), UML modeling tools are useful for formal model definitions. These include tools like Umbrello, Dia and Magic Draw. For the development itself there are some frequently used IDEs, like Netbeans and Eclipse, although many developers prefer to use functional text processors, like vim or emacs. When very specific or isolated environments are needed, development can be done using virtual machines. The preferred option is VmWare , as ACS distributes VmWare machines along with its major releases. Finally, bug management systems are needed to provide feedback to the ongoing projects, or even assign essential system administration tasks. These have evolved from Bugzilla, to more recent work with Flyspray and Jira.

While some of these tools and platforms are essentially ESO standards, adopted later by ALMA (e.g., Yahoo! Messenger, TWiki, and CVS), CSRG makes use of own standards as well. The most important case is git, a distributed version control system, which is used by CSRG to keep track of internal projects. Following this example, git is currently also being used informally by some ALMA software engineers in several ALMA development phases (e.g., development during HEAD freeze, and branch merging). The usage of non-ALMA-standard operating systems in CSRG, like Ubuntu and Fedora, has also ended up with contributions to ACS, like patches to extend its compatibility across several platforms.

Interaction and collaborative work is mostly restricted by the CSRG's characteristics as a students organization. Therefore, development cycles are not always predictable, and working hours differ widely from usual office hours. Also, the fact that not all developers/students have an advanced technical level, complicate sometimes the initial work (training) process. Although this may require some adaptation efforts by the supporting organization, a high quality work can be achieved by students, and they are actually able to interact side by side with professional engineers, maintaining a strict schedule.

5. PRACTICAL EXPERIENCES AND EXPANSION OF THE MODEL

In section 4 we have described in general terms collaboration approaches, which we will complement in this section with practical experiences. We will now describe relevant CSRG projects according to main development areas: ACS products, learning platforms, research projects and publications, undergraduate and master theses, and summer internships.

5.1 ACS products

The first main interaction point with ALMA was the development of ACS applications. At the beginning, they were only nice-to-have applications, which would have been useful for ACS, but not critical. Some of them are even more oriented towards the ACS open-source community than of direct application for ALMA. Also, most of these projects allowed to explore new alternatives in terms of technologies and strategies.

The first ALMA-UTFSM venture was the ACS Packaging project, started in July of 2004. This project was based on the fact that ACS was (and still is) a huge block of binaries that has to be installed from a tarball. The aim was to provide an RPM installation, which is the RedHat package standard. An automated procedure to build an RPM from the ACS binaries was provided, and although this installation type has been never used afterwards as an official distribution medium, this was an important step to get involved with the framework's overall design.

In early 2005, and as part of a summer internship at the La Silla site, a first official ACS tool was implemented. The CDB Checker was a validation tool for the ACS configuration and deployment XML database. Although it was first seen as a barely useful application, it became one of the most used ACS tools until today, specially as the configuration database became more and more complex. Further ACS tools included the Error Browser and Editor (a graphical configuration tool for ACS error definitions), the Alarm Configuration GUI (a configuration utility for ACS alarm definitions), and the Sampling System GUI (a trending tool to plot hardware properties through the ACS sampling system API). All of them were developed in a similar way, starting from the requirements given by the ACS team in Garching or the ALMA Computing Group in Chile, and working mainly individually, until a usable application could be iterated in conjunction with the ALMA counterpart.

Besides the official ACS products, some general purpose contributions were made to ACS. In 2007, the ACS Forge Initiative (AFI) was started. This project explored different alternatives for a revision control and software development management system, with the goal of becoming a centralized resource for managing projects, communications, code and tests of contribution modules developed for ACS by the community. The project concluded in late 2009 with a working platform and the community is now starting to use it. Since 2009, some effort has been put into the port of ACS, which runs traditionally on Linux distributions, to MS Windows (see[8]), mainly because of the need of having a broader support of platforms and with the direct request coming from projects interested in using ACS on Windows, where most industrial control tools are available. Also, recently the group has been working on various code generation applications, in order to provide a simplified developer experience on top of the ACS framework. Thus, prototype tools for generating ACS code starting from UML class diagrams and state machines have been implemented (see[9]). Recent ACS contributions have been described in.[10] As these are projects that mainly explore new areas, requirements were mostly defined in conjunction with ACS developers, and developed unilaterally, with a close feedback interaction.

5.2 Learning platforms

As one of the main missions of the group is to learn about and create expertise on distributed systems, the ACS platform has been a recurrent example platform for creating applications with learning purposes. While for some people it is exciting to get into theoretical concepts, for a majority something that moves and that you can touch certainly motivates. Thus, one of the most important steps of the group's kick-off was the donation of a Lego Mindstorm kit by La Silla IT Manager Flavio Gutiérrez. With this kit, the Scientific ACS-based Remote Lego Telescope (ScarLet) project was initiated in March of 2005. The idea was to use the Lego kit to build a telescope toy model and control it remotely using ACS, through a functional GUI.

In 2006, a continuation of the ScarLet project was launched, but this time as an ACS learning platform for training purposes. The Hardware End to End Example (H3E), aimed to build a complete example demonstrating ACS controlling a hardware system connected to a PC. The most important deliverable was a documented and reproducible design and development, allowing ACS newcomers to perform self trainings. Although some new team members were trained using H3E , the project was later on abandoned as a learning method, mainly because it did not promote team interaction during the learning process, and was therefore not very efficient.

Alternatively, the practical part of the yearly ACS Workshops was very well exploited as a complete development and integration experience. These workshops were organized since 2004 by ESO and the ACS community, in order to get more people involved in the framework's usage, and to concentrate in a short period of time training sessions for all newcomers. In mid 2007, the 4th ACS Workshop was the first one to be organized at UTFSM, featuring a 3-days hands-on project. The project was called UTFSM Observation Software (UOS), and aimed to develop a more or less complete example of an observing software for a virtual observatory, divided into 6 modules (each one developed by a 3 to 4 persons team) to be integrated, and including hardware elements like a CCD camera and an amateur telescope. The only previously defined parts were the interaction of modules and their public interfaces. In parallel to the development teams, an Integration and Testing team was in charge of maintaining a healthy and working code repository, and performing the essential final integration, including testing and final "acceptance". The course instructor was in charge of guiding the development and maintaining the project's schedule on time. While learning about the ACS architecture, language transparency and interoperability, and distributed deployment, this course also allowed participants to experience a real project development cycle, with real pressure due to the module dependencies and the fixed deadline. UOS was used in a second opportunity at the 2008 ACS Workshop's basic and advanced track.

An updated basic track was introduced at the 6th ACS Workshop in late 2009. The Maintenance And Repair System (MARS) simulated a distributed space-ship maintenance system, controlling a set of Lego robots that had to move to precise locations on a grid (see Figure 6), triggered by (simulated) alarm sensors or maintenance schedules. Besides the same concepts included in the UOS system, MARS introduced some advanced concepts like the ACS alarm service. Participants at this workshop came from many organizations, including ESO, NRAO, NAOJ, Universidad Católica del Norte and Universidad de Concepción.

5.3 Research projects and publications

From the CSRG point of view, research projects are among the most important activities, with the objective of exploring novel technologies and complex designs for new system infrastructures. As mentioned before, the first major project in which the group got involved was the Generic Telescope Control System. The gTCS was a concept being discussed since the 1st ACS Workshop, and considered a major challenge. The purpose was that to provide a generic design for a TCS that could be shared among a large variety of (amateur and professional) telescopes, only reimplementing the lowest level of hardware interaction. The motivation was, as of ACS itself, that of making available on top of the ACS services a real astronomical domain specific framework, avoiding duplication and allowing reuse: this would have been a strong reason for other astronomical projects to adopt ACS. Basic models like the ScarLet project (see section 5.2), and later CSAT, were first experiences towards this

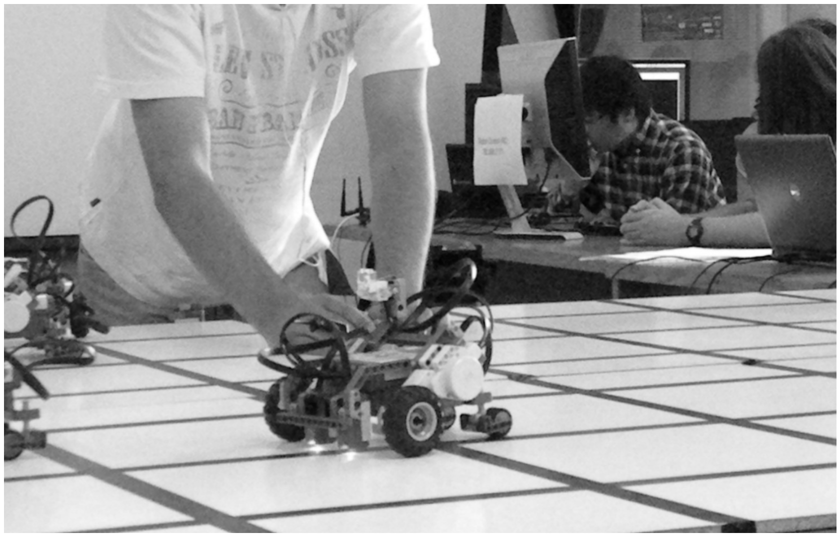

Figure 6. Lego robots during the 6th ACS Workshop.

goal. The CSAT (Control System for an Amateur Telescope) project considered the implementation of an end-to-end telescope control system to manage observations with a NexStar 4SE amateur telescope (see[11]). Far ahead its initial goal, CSAT was actually proved to control different types of amateur and semi-professional telescopes, with both equatorial and altazimuth mounts. Testing sites included teaching observatories of Universidad Católica de Chile (Observatorio UC, Santa Martina, Santiago) and Universidad Católica del Norte (Cerro Armazones Observatory, close to Antofagasta). As parallel projects, to develop other related areas, the Repackaging ACS for Embedded Systems (RAES) and Alternative Real-Time Platform for ACS (ARPA) projects were carried out. A gTCS design architecture request-for-comments was launched in late 2008, and a general implementation prototype was finished during 2009.

In another area, since 2009, the Artificial Intelligence in Astronomy (AIA) group is working on different projects, researching and developing methodologies and solutions for complex problems arising in the astronomy computing field, particularly using Artificial Intelligence heuristics. This sub-group is currently working on the observations scheduling problem, taking advantage of the ALMA Scheduling subsystem position that is being financed by NRAO at the University, allowing a half-time work on internal research projects (see section 5.6).

Naturally, research projects also produce publications. Besides periodical internal technical reports, CSRG has been mainly publishing in national and international conferences, in computer science and astronomical engineering areas. First projects were presented at “Encuentro Linux”, the largest open-source related annual event in Chile, and “Reunión Anual de la Sociedad Chilena de Astronomía (SOCHIAS)”, the annual meeting of the Chilean astronomy society.[12] However, these were more about getting known, and spreading the overall group’s activities. Specific relevant scientific conferences were attended since 2008, after major results had been obtained in the ongoing projects. In the computer science area, papers have been presented at “Jornadas Chilenas de la Computación (JCC)” (annual meeting of the Chilean computer science society) and “Conferencia Latinoamericana de Informática (CLEI)” (largest Latin American informatics conference); while in the astronomical instrumentation field, various editions of “Astronomical Data Analysis Software & Systems (ADASS)” and SPIE conferences have been attended.

5.4 Undergraduate and master theses

Along with conference publications, CSRG internal research and development projects have also produced an important amount of undergraduate and master theses. Normally, theses have been a starting point for some projects, that were afterwards further developed by other students. Theses are also a significant point of technical interaction with faculty professors, and with ALMA software engineers.

At this point, we have to clearly distinguish between undergraduate and master theses*. While the first ones devote to practical engineering work and require a concrete working implementation, the latter ones are much more theoretical and explore new knowledge in the context of a major research area. Although master theses require at least one publication at a prestigious international conference or journal, in our case also undergraduate theses have produced a good amount of technical publications. Naturally, on the technical interaction level with ALMA software engineers, it is much easier to find proper topics for undergraduate theses, as they can provide concrete products in a more or less fixed time frame.

Various working approaches have been used in order to integrate theses developments with collaborators, or use external technical resources. For example, remote hardware has been used for real-time operating systems related work, as specific systems are not easily available at the university. Thus, remotely shared servers can be accessed over the Internet. A good way to facilitate the technical work with partners at ESO and NRAO have been visits to the ALMA development sites; not only to discuss more easily about specific topics, but also to share some time at a real engineering work environment. Also, individual initiatives have been hosted by the group; in this case, interaction with ALMA developers is limited, but new fields not immediately related to the project can be explored.

Since 2008, nine undergraduate and master theses have been produced inside CSRG, and several others are under development.

5.5 Summer internships

Since the very beginning of CSRG (and even before), summer internships have been a keystone of the group's development and motivation. Until 2007, internships were concentrated exclusively at the LPO La Silla site, and were normally about exploring and prototyping new technologies which could be eventually used at the observatory. For example, a hard-drive diagnostic and error alert tool on top of SMART was built during 2006. Between 2002 and 2007, a total of 13 summer jobs were performed at the La Silla site by students related to CSRG.

In 2008, student internships at La Silla were canceled due to the observatory's resource reduction plan. Nevertheless, that same year the group managed a total of 10 summer jobs at different institutions: NRAO, ESO, AURA (Gemini South Observatory), Universidad Católica de Chile and Universidad de Valparaíso. This included the first international internship at NRAO facilities in New Mexico. Together with a remote internship in Valparaíso with ESO headquarters, these were a first successful attempt on working as part of ALMA Software development groups, and producing important contributions to the project code. Both of them used similar collaboration approaches by working as part of geo-temporally distributed teams. In particular, interaction through TWiki was very relevant to maintain track of the work from both the student's and the supervisor's side, and was also actively used for design discussions.

During summer of 2009 and 2010, a total of 15 internships, including three international jobs at ESO and NRAO were performed. Other supporting organizations included AURA, ESO (Paranal Observatory), Universidad Católica de Chile and ALMA, featuring first internships at the observatory site in Northern Chile. Also, every year some internal summer jobs are performed at UTFSM, in the CSRG laboratory, mainly including further development of ongoing projects.

5.6 Discussion and further expansion of the model

Among common characteristics between developed projects, we can distinguish the usage of the TWiki platform as centralized endpoint for project discussion and coordination and periodic work-logs. Specially in highly collaborative distributed projects, this platform is used as an elemental communication tool. Also, all products developed by CSRG itself are released under open-source license, and allow therefore further usage and contributions by third parties.

We can conclude that projects that are developed with a clear start and end point during summer internships tend to produce a more efficient work, also because of the 100summer vacations. In terms of research areas, the

*The exposed distinction between undergraduate and master theses is specific for the UTFSM definition, and may or may not be the same of other institutions.

main advantage of the work done by students and professors inside a university, compared to ALMA (or other projects') developers, is that they are free to test different alternatives and play with new technologies, as they are normally not dealing with critical software parts that have to be working in strict schedule. This produces eventually benefits for the supporting organization, but not necessarily in the short term.

Since 2006, former CSRG members started working as professional engineers for the ALMA project. As the site infrastructure in Chile was still under constructions, most of them were sent to development sites in Germany and U.S.A. In 2009, a first former team member was hired for a position abroad, as part of the ALMA Computing team at NRAO, in Socorro, New Mexico. These positions represent a major binding between the group at UTFSM and the industry, as most of these former members maintain an active interaction with the group. They are therefore a continuation of the group in the professional world. In 2009, for the first time a person was hired at the university, working half-time for ALMA and half-time in research projects inside CSRG. This new, reversed model resulted in a very good approach, because it brings inside the university a project team member closely involved both in complex development for the project, and research projects in the university, providing a closer connection with the ALMA software engineers inside the group. Currently, there are two positions of this kind: one financed by NRAO, and another one by ESO.

Finally, this model is being adopted by other universities in the software and electronics areas. Particularly, there have been successful collaboration projects with Universidad Católica del Norte (UCN), Universidad de Valparaśo (UV) and Pontificia Universidad Católica de Chile (PUC), and a small development group is starting to work at UCN, mimicking the UTFSM experience. The UCN campus in Antofagasta will host the ACS Workshop 2010, organized in conjunction with ALMA and UTFSM. It is important to notice that engineering for astronomy is growing to a strategic area in some Chilean universities because of the huge impact that these international facilities are having in the country's engineering market. But inter-university collaboration is essential to meet the diversity of specializations that are currently required by observatories.

Also, other technical areas inside the ALMA project are getting interested in this model, as successful training process for engineers going to work in astronomical observation facilities: students participating in these projects get an on the job training that will allow them to be immediately productive as hired later on. After first software summer jobs at the ALMA Operations Support Facility in Chile in early 2009, the Assembly, Integration and Verification team has featured several summer jobs in the electronics engineering and technician areas starting in 2010.

6. CONCLUSIONS

We have presented the overall ALMA - UTFSM collaboration, including all elements we consider relevant in order to understand and further apply and extend the model. The core of this model relies on a self-organized students initiative, actively collaborating with an observatory's engineering staff. The following key elements have been identified:

- **Supporting organization:** Motivated counterpart for the specific working area.
- **Students:** Highly motivated and enthusiastic, selected both by technical (hard) and social (soft) skills as part of a main development team.
- **Faculty members:** Supportive and involved in the students group's projects.

It is very important to point out that at the beginning of this collaboration both on the side of the students and of the supporting organization there were preexisting high-quality teamwork experiences: this has constituted the groundwork for the creation of the synergy between the two teams. This fact has been identified as a critical element for the success of this experience, along with an optimal communication between all involved parties. Communication has to be frequent, periodic and redundant. Also, for any new project that wants to be developed inside this collaboration, two basic requirements have to be satisfied: a committed contact point inside the supporting organization, and a topic relevant and interesting for both parties.

Lessons have been learned by both parties involved in this journey, and we can recognize that there is an opportunity window for other collaborations based on a similar model. We see that there is a growing interest on our experience coming from several directions and we want to ensure through this paper that new enterprises can profit from our learning process. The experience of participating in such initiatives is extremely enriching for students, and has been proved to be a viable on job training process for interested organizations, to receive immediately productive engineers.

We can further conclude that early participation in upcoming major scientific projects is fundamental, and allows to anticipate -in our case about 5 years in advance- the creation of a highly competitive on-site support team. This has been a very successful experience for ALMA, and it has been recognized that multidisciplinary inter-university collaboration is aligned to the organizational structure of current international projects' development. For future astronomical projects currently in early development phase, like the European Extremely Large Telescope (E-ELT) and the Large Synoptic Survey Telescope (LSST), it would be now the right time to consider this important opportunity.

ACKNOWLEDGMENTS

The authors of this paper have all been actively involved as part of the ALMA - UTFSM collaboration. Some of them participated (and graduated) as outstanding members of the university group, while others acted as essential counterparts at ALMA/ESO since the very beginning. Most of them are today working as part of ALMA development teams around the world and/or doing postgraduate studies.

REFERENCES

[1] Chiozzi, G., Gustafsson, B., Jeram, B., Plesko, M., Sekoranja, M., Tkacik, G., and Zagar, K., "CORBA-based Common Software for the ALMA project," in [*Proceedings of SPIE 2002*], *Advanced Telescope and Instrumentation Control Software II* **4848**, 43–54 (aug 2002).

[2] "Computer Systems Research Group (CSRG)." `http://csrg.inf.utfsm.cl/` (may 2010).

[3] "6th ACS Workshop 2009." `http://acsworkshop.inf.utfsm.cl/` (may 2010).

[4] "ALMA-UTFSM Group." `http://alma.inf.utfsm.cl/` (may 2010).

[5] Levy, P. E., [*Industrial/Organizational Psychology*], Houghton Miffin (2003).

[6] Gonzalez, V., Mora, M., Araya, R., Arredondo, D., Bartsch, M., Burgos, P., Ibsen, J., Reveco, J., Saez, N., Schemrl, A., Seplveda, J., Shen, T.-C., Soto, R., Troncoso, N., Zambrano, M., Barriga, N., Glendenning, B., Raffi, G., and Kern, J., "First year of ALMA site software deployment: Where everything comes together," in [*These proceedings*],

[7] "Programa de Iniciativas Estudiantiles Académicas (PIE>A)." `http://www.piea.usm.cl/` (may 2010).

[8] Tobar, R., Menay, C., Avarias, J., and von Brand, H., "Porting a large distributed control framework to the Windows Platform," in [*Proceedings of Jornadas Chilenas de Computación*], *XIII Workshop on Parallel and Distributed Systems (WSDP)* (nov 2009).

[9] Troncoso, N., von Brand, H. H., Ibsen, J., Mora, M., Gonzalez, V., Chiozzi, G., Jeram, B., and Sommer, H., "A code generation framework for ALMA Common Software," in [*These proceedings*],

[10] Chiozzi, G., Jeram, B., Caproni, A., Sommer, H., Sekoranja, M., Schwarz, J., Cirami, R., Yatagai, H., Avarias, J., Hoffstadt, A., Lopez, J., and Grimstrup, A., "ALMA Common Software Status and Development," in [*Proceedings of 12th International Conference on Accelerator and Large Physics Control Systems (ICALEPCS)*], *Advanced Telescope and Instrumentation Control Software II* (oct 2009).

[11] Tobar, R., von Brand, H. H., Araya, M., and Lopez, J., "An amateur telescope control system towards a generic telescope control model," in [*Proceedings of SPIE 2008*], *Advanced Software and Control for Astronomy* **7019** (jun 2008).

[12] Araya, M., Avarias, J., Mora, M., and Tobar, R., "ALMA Common Software - UTFSM Group," in [*Proceedings of VI Reunión Anual Sociedad Chilena de Astronomía (SOCHIAS)*], 58–+ (2007).