GLORIA is funded by the European Union 7th Framework Programme (FP7/2007-2013) under grant agreement n° 283783

GLObal Robotic telescopes Intelligent Array for e-Science

Documenting of experience existing within the GLORIA collaboration

CODE: DEL-060

VERSION: 1.0

DATE: March 31, 2013

Authors:	Ronan Cunniffe (CSIC)
Collaborators:	Miquel Serra-Ricart (IAC) Luciano Nicastro (INAF) Martin Topinka (UCD) Lorrain Hanlon (UCD) Jan Strobl (FZU-CAS) Michael Prouza (FZU-CAS)
Revised by:	Raquel Cedazo (UPM) Mercedes Grijalbo (UPM) Francisco Manuel Sánchez (UPM) José María Sebastián (UPM) Fernando Serena (UPM)
Approved by:	Raquel Cedazo (UPM) Francisco Manuel Sánchez (UPM)

Distribution List:

Name	Affiliation	Date

Change Control

Issue	Date	By	Change Description
0.1	20/02/2013	rcunniffe	Initial version
0.2	20/03/2013	rcunniffe	Re-structured based on early q. responses
0.5	27/03/2013	rcunniffe	First circulated version.
0.9	29/03/2013	rcunniffe	Final responses received & corresponding text written & checked.
0.95	1/04/2013	rcunniffe	Final editing.

Index

1. Introduction

1.1. Purpose

We recognise that one of the most valuable contributions the GLORIA collaborators can make is to share the experience that we have gained in how to plan, build, operate and maintain robotic telescopes – and usually experience in how *not* to do those same things. This document is a synthesis of that experience, organised by topic, and is meant for novice roboticists to learn from – and for experts to smile or wince at, as appropriate.

This document was born of a questionnaire that was passed around the GLORIA partners, trying to elicit the *details*, rather than generalities, on the grounds that this document would be assembled by those already familiar with robotic astronomy, but missing the wealth and depth of detail available from such a broad collaboration. The questionnaire is included in Appendix A.

1.2. Structure

The structure of this document follows that of the questionnaire: From startup, through technology choices and installation, site choice and equipment maintenance, data handling and processing, and disasters and disaster recovery. The last section is advice to a novice, and raises the obvious question: for whom is this document written? The simplest answer is that it is written for ourselves as we were starting down this road; a mix of experienced astronomers learning telescope maintenance, software and hardware engineers learning astronomy. We needed details, not a handbook of project or operations management (actually, that might have been useful too....). We needed to know what to think about, what to pay attention to, in this rapidly changing corner of astronomy.

On which note: the state of the art is changing *very* rapidly in robotic astronomy, and the difficult technical challenge of a decade ago can (nearly) be bought off-the-shelf today. The exact details of our experience may soon sound as quaint as talk of floppy disks and telephones attached to the wall, but a great deal is about the shape of a successful astronomy project, and that is what *we* give priority to.

2. Startup

2.1. Site and telescope

All the GLORIA collaborators chose the same pattern for their project - to build a new instrument at an existing site – and the reasons given were largely the same:

- technology improves in astronomy just as in any other field; a new instrument is likely to outperform an existing one.
- when robotic astronomy is the answer, the question often involves an unusual combination of ideas and technologies, and a poor match with existing instruments.
- existing telescopes tend to be either busy or dead. They are operated until forced to stop, which is often due to the need for an expensive repair.
- for non-robotic installations, minor technical problems are often “fixed” with a human workaround, because the underlying cause are either not properly understood, or too expensive to properly fix. Robot systems cannot do this, so robotizing an existing system means identifying, understanding, and fixing all such technical problems first.
- designing and build a new instrument is predictable. Robotising an existing telescope is almost

guaranteed to offer unexpected costs and delays, unless the people involved know the telescope very well at the engineering level.

As for building at an existing site, *every* telescope in GLORIA has a designated human nearby, with multiple lines of communication in case of disaster. It is a painfully learned lesson that robotic telescopes are not viable without either human support on site, or an order-of-magnitude increase in engineering cost. The design focus has universally been – across the range of GLORIA instruments – that simple problems can be dealt with by remote technology, but that the reaction to anything major is contact a nearby human, even if just to make sure the dome is closed and everything unplugged.

2.2. How long does it take?

It varies. A lot. From project beginning to robotic operation, the range within the GLORIA partners is 1-5 years.

The biggest predicter of duration was team experience. FRAM (25cm) was built by people familiar with all the hardware and software they were using and was built and installed in 1 year – while BOOTES-3 (60cm) also took 1 year, despite being a much bigger installation, because it is a copy of BOOTES-2, which took (more or less as planned) 3 years to reach robotic operation.

There appear to be two distinct strategies: *commission-then-robotise* and *robotic-from-the-start. Commission-then-robotise* is slower, but nicely separates mechanical, electrical and optical problems from the robotic software effort, so that by the time the robotic software takes over, the telescope behaviour is known, so any new odd behaviour can be immediately attributed to the software.

Within GLORIA, *robotic-from-the-start* was mostly attempted only by teams with prior robotic experience, and was very stressful for those attempting it with their first instrument. Of course, commission-then-robotise is only realistic when the telescope is nearby, a lot harder if it is on the other side of the planet. The ideal (and rarely practical) strategy would be *commission-then-robotise-then-transport*, or something like:

1) install the telescope next door

2) work the bugs out of the hardware

3) install the robotic software stack

4) work the bugs out of *that*

5) carefully take the system apart, ship it to the real location, and re-assemble it.

6) check you've re-assembled it correctly...

The FRAM team did manage to do this. The BOOTES team *almost* managed it by building BOOTES-3 as a twin of BOOTES-2.. Almost: the new site had a different dome mechanism – which promptly caused problems! Likewise, TAD and Montegancedo are deliberately nearly identical, with Montegancedo sited (not accidentally) on the same university campus as its software developers.

The common thread across many of the GLORIA telescopes was: nothing works the first time, and the longest projects were those that combined technical ambition with - by the team's own admission - insufficient testing before deployment. Wise readers should recognise that the GLORIA partners are not obviously stupid: “insufficient” means “underestimated”, and “underestimated” means “the reader is about to make the same mistake”: the required development and testing effort is *strongly* non-linear with technical ambition, and (in academic environments) a delay can mean disproportionate chaos when a critical graduate student has to move on.

This underestimation tended to happen in software development:

- because hardware choices tended to be conservative
- because software was where the most technical ambition lay
- because writing software that acts in the real world is unexpectedly (to a novice) hard to do.

2.3. Site:

The GLORIA telescopes are situated in quite a wide range of environments, from desert mountaintops to coastal plains. A number of telescopes have resorted to air-conditioning in the dome against overheating – usually the computers are the most vulnerable. Mostly, the environment simply leads to accelerated degradation – rusting of metal parts, perishing of rubber seals, dust on the optical elements, or turning lubricants into sludge – and well-engineered systems can ignore them for extended periods of time. One response specifically mentioned pollen (or some sticky organic particles) making the mirrors uncleanable.

The most cited worry was lightning – Watcher and BOOTES-1 were both destroyed by it, and TAD was engineered from the beginning to resist. Certainly in the BOOTES-1 case, the dome was not directly hit, but the power distribution system on the host site was, and the effect was enough to destroy everything electronic in the dome. A UPS to act as a sacrificial 'fuse' (not just as emergency power) is a common response – circuit breakers are usually rated to 6kV, i.e. anything that *isn't* lightning.

2.4. Hardware:

> *"Do not buy cheap solutions."* (FRAM)

> *"One component failure is usually enough to kill a robotic telescope for at least one night. How many such components in the system? And how many nights do you intend to lose?"* (BOOTES)

This was the most emphasised point: most technology is simply not engineered to the standard needed for unattended operation – for manufacturers it is easier (i.e. cheaper) to get a *mostly* reliable product out the door (and hey, if it glitches, just turn it off and back on again). Of course, high cost is not a proof of high quality, and the only definite advice to give here is: find experts and ask their specific advice. Are we not experts? Yes, but advice would have to be specific to the project goals.

When asked about technology upgrades and replacements, two components – mounts and domes – were easily the most significant causes of problems (or, at least, the components whose problems were most memorable). See Disasters...

More than one of the GLORIA partners started with under-sized or under-engineered telescope mounts, and had to replace them. Others cheerfully admit they deliberately over-sized their mount choice and subsequently expanded their instrument (e.g. a secondary instrument riding piggyback) to make use of their investment. As always, the most expensive solution is the one that doesn't work.

Many other devices such as CCDs, filter wheels, focusers, etc., were also likely to have been the subject of upgrades, but the questionnaire responses suggested that mount changes are in a different class, both in cost and impact – it was the *only* part of the system that was repeatedly raised as critical to get right. This is not unexpected, as a mount change requires the complete disassembly, rebuild, and recalibration of the whole instrument: psychologically a mount change is a change of era. Remarks on other technologies (optical elements, cameras, filters, etc.) clearly considered them as commodities to be chosen by price and scientific goals: "The OTA was the very best we could afford, as was the camera", and no details or comments were made.

Some interesting comments about telescopes:

- Open truss telescopes suffer less wind-induced vibration than solid barrel designs, and this can matter. (Watcher, BOOTES-3)
- One site (FRAM) replaced its original Cassegrain telescope with Schmidt-Cassegrain, due to repeated optical alignment problems. The new telescope had poor UV transmission (which was a significant drawback), but at least it remained in focus.

2.5. Sensors

Every GLORIA telescope has at least one remotely accessible camera, and many have two (inside and outside views). There is a strong preference for cameras designed for security purposes even for cameras inside the dome, rather than webcams despite significantly lower cost and frequently higher resolution – robustness and device independence are the features desired. IR lights are also a feature, since domes can easily be too dark for a small lens.

The most important sensory information is weather. Most of the GLORIA telescopes are situated in larger astronomical sites, with network-shared weather information. Such information can be relied on to be accurate, but makes the telescope dependent on the network. So, again, most of the GLORIA telescopes have access to (and use) such information, but also have their own sensors and use both: their own sensors to avoid an automatic shutdown if the network is interrupted; the network service to verify that their own sensors are working. *Every response described two separate sensors for detecting rainfall.*

2.6. Tools and equipment on site:

Interestingly, the responses varied from “Everything could come in handy :-)” to “We try to remove every tool that it is not necessary”. Telescopes on existing observatories tended to emphasise what was available on the larger site, rather than what was kept in the dome. One response advocated each team member being responsible for having their own tools, and not assuming that there was anything at all on site. Knowing what tools to bring then becomes important, and having a checklist (i.e. a requirements list for wrenches, allen keys, large and small screwdrivers, etc.) for the site.

2.7. Equipment and cabling:

Mounting equipment, cables, power-supplies, etc. can turn a telescope into a badly-decorated Christmas tree, with cables lying across the floor waiting for the feet of the unwary, and small domes don't leave much space to avoid them.

The commonest solutions were:

- we know where the cables are, and we don't trip
- cable ramp on the floor
- hole in the floor (either a false floor, or a genuine pass-through to room below).
- computer cabinet mounted directly against the pillar
- forbidden sector of the dome (e.g. a desk or other obstacle to stop people walking across exposed cabling).

The solution used seems to have been determined mostly by whether or not cabling had been specifically considered early in the project.

3. Software:

Almost all the GLORIA telescopes are still using the software with which they were originally commissioned – the software is far more permanent than individual pieces of hardware – and there is a corresponding sense that the software is *the* major investment that needs to be done correctly the first time.

Of the 17 GLORIA telescopes, 13 are using the RTS2 observatory manager software, chosen for a wide variety of reasons, but chiefly because it existed – using RTS2 meant not having to build their own. It should be noted that one GLORIA instrument was initially implemented using RTS2 and then switched *away* to an alternative (ACP, on Windows), but this was away from one off-the-shelf solution to another, not to an internal project.

3.1. Data storage, transport and processing:

Onsite storage didn't appear to be considered a problem, although two responses had negative things to say about NAS (Network Attached Storage) devices. Most sites have sufficiently good internet connections that transporting a copy offsite was not difficult (BOOTES-3 and Watcher may be the only exceptions). Redundant copies seems to the answer, and with increasing Internet speeds, the solution is usually to have one of those copies is usually back in Project HQ, and any data on site can be deleted as and when required.

The most common pattern seems to be to do basic image reduction work (e.g. astrometry) on site, and largely automated, but a great deal of science is done after retrieval. A number of projects originally intended to do more processing on site, but the difficulty of writing automatic processing pipelines, and the advent of faster Internet links have somewhat derailed the need for this.

4. Operation & upgrades

“Later on, the telescope was variously upgraded, repaired, etc. Not much of the original design actually remains, except the dome.” (FRAM)

A number of the GLORIA partner telescopes have been operating for ten years, and there is a common pattern of gradual technology replacement, to fix behavioural problems, to replace failed parts, or in support of expanded scientific programs. In other words: every problem or failed part is to be considered an opportunity to improve the system, funding permitting. In many cases, the real aim of the long-term members of the teams is scientific, so evolution slows drastically when no serious problems remain – if it ain't broken, don't fix it. One response indicated that the team was assembling the pieces for a complete replacement.

4.1. Emergency planning:

Each response mentioned “UPS” *in the first sentence*, and some sites have redundant UPSes. Redundant weather sensors were also common, and people were at least thinking about an active failure alert by SMS.

4.2. On site support:

Nobody seems to think it's a good idea to work without local support – all the responses were emphatic, with examples of exactly how they made use of that support: moving stuck mounts, PC failures, etc. None of the responses indicated that an expert was necessary, just somebody able to reset things, and report on whether something was running or not.

4.3. Maintenance:

Almost no preventive maintenance was done – the efforts described were all geared towards responding to failure – usually by fixing the failed part (e.g. cleaning shutters, replacing hard disks) and if repeated failures occurred with the same part, replacing it with something more robust. This is largely due to the time pressures on personnel, not because it was considered an ideal approach.

Some comments:

- many technologies are unable to cope outside office conditions, e.g. shutters, circuit boards.
- hard-disks don't seem to like high altitude – SSDs should now be considered a very wise investment
- keep spares onsite of critical PC things – hard-disks, fans, power supplies; the hardware cost is definitely marginal compared to the telescope management.

5. Disasters:

The Number One Disaster: Dome open during heavy rain. There is clearly something memorably horrible about looking at a webcam image of an open dome..... with rain running down the webcam window (or in the case of TAD and BART, *snow*). More importantly, the various causes of these dome disasters were:

- misconfiguration – high humidity caused the dome to *open* instead of close.
- electrical fault - one half of a clamshell opened spontaneously, and all open/close toggles just changed *which* half was open.

- sensor failure that didn't trigger a fail-safe condition in the dome controller.

Of course, *you* would have spotted all of these design weaknesses without fail, right?)

What is truly remarkable about these occasions is how well the installations actually *survived* them. Camera shutters (and sometimes the cameras themselves) were destroyed, but telescopes, mounts, and much of the equipment in the dome continued to work afterwards. This may be very much the result of the harsh requirements for robotic astronomy equipment anyway – anything that has lasted long in a dome is already not-the easiest thing to destroy.

Other disasters (or near disasters – memorable if not always destructive):

- collision between telescope and roof
- collision between telescope and mount
- telescope tried to set fire to the building by reflecting sunlight onto the underside of the wooden roof.

What telescopes clearly cannot survive at all is lightning through the electrical system (either power or network). That has killed at least to two GLORIA telescopes (as well as obliterating all data on site), so trust us on the UPS.

6. The Grand Summary: Advice to a novice:

The last question asked of the GLORIA partners was what advice would they most liked to have received starting out, based on what they knew now. Most of the following points are verbatim comments from those who have learned lessons the hard way: the seem obvious until you fall into the same trap.

6.1. Site:

- Install your telescope in a site where other installations are running with on-site support.

6.2. Technology:

- Do not buy cheap solutions. You need reliable components that will endure long term robotic operations, and such components are not cheap. Thus take this fact into account when putting together your budget. Look elsewhere for proven components, discuss with your colleagues what CCD cameras, telescopes, mounts, enclosures, etc. are the reliable ones before you realize the purchase.

- The most expensive solution is the one that is *nearly* good enough.

- Try not to ship without comprehensive aggressive testing of each component and the whole system. If shipping to the far side of the planet, *try really, really hard*!

- Have a way to remotely reset every piece of hardware in the system. For network-aware devices, choose ones that can be remotely reset and can save and restore their configurations – the engineers who built them are thinking the same way you are. For everything else, use controllable power-strips – they're less expensive than plane tickets or car fuel, not to mention the lost time.

- Never trust the claimed capabilities of a component if not proven with actual data.

- Have your own sensors and cameras. You need cameras covering inside and out. Any critical sensor (e.g.

weather) data should come from at least two independent sources. Consider connecting such data to your dome electronics directly. Log your sensor data. Log *everything*.

6.3. People

- Never get involved in projects or groups were the responsibilities are given to people without proven expertise. Be happy with a project with limited but solid capabilities rather than one that pretends to be able to do many things. A small project requires a small number of people, but motivated and capable. Always ask yourself: who is going to do what? The answer will guide your choices.

- Agree software interface specifications before any code is written. Push for it even in the face of strong resistance – this is the only way to develop the pieces in parallel, to get most disputes out of the way early about which tasks belong in each logical piece or layer, and then if changes need to be negotiated later, at least everybody is speaking the same language.

6.4. Processes

- Don't build a collection of parts and hope they will become a working system. Integration is its own discipline and requires a specific effort, and the planning is best started early. For hardware, it is highly recommended. For software development, it is *mandatory*.

- You need formal procedures and documentation more than you think – and they must be kept up to date. One person remembering things works ... until it doesn't, and then it's a real problem.

- Can a new team member *read* themselves into familiarity, or will they have to be *told*? By multiple people? Some of whom have now left the team?

- Somebody announces they're leaving the team, and you panic: “Get them to write everything down first!”

- Ok, lightning has just destroyed all the computers on site, but it's just a matter of buying replacements, because *of course* you have a (recent) offsite backup of the configuration.... right?

- If you can't do any remote diagnosis, every problem guarantees two trips: the second is to fix the problem, the first is to figure out that you brought the wrong equipment. Consider remote diagnosis as a basic design goal from the start, and save yourself the stress of having to retrofit it later. If relying on a local human to diagnose problems, consider the advantages of not dragging them out of bed *every* time.

7. Appendix A: The questionnaire

Deliverable 4.5 asks us to document the knowledge and experience of those in the GLORIA collaboration. The goal is to organise what we have learned about successfully (and perhaps not-so-successfully) designing, building, and operating a robotic instrument. *If you could go back in time and advise your earlier self, what would that advice be?*

So, what follows is a set of guiding questions - answer as many as are appropriate. Please emphasise the *details* of your experience rather than generic robotic astronomy – that will be added into the final integrated document (think "telling a story" rather than "writing a textbook").

Obviously, this task is being asked of very busy people, but the following approach should work well:

1) Spend 10 minutes reading through the questions *right away*. Maybe split the job by section to different people?

2) Make notes of anything you think of immediately.

3) Allow the questions to cook slowly for a day or two.

4) Write longer answers to the questions where you have some experience to add.

Please do answer as many questions as apply – your experiences don't need to be *unique* to be valuable. It would be *very* interesting if most/all projects under/over-estimated the same thing.

1. Startup:

1.1: How long has your project been running?

1.2: Why did you start, and have the goals changed since then?

1.3: Did your project involve an existing telescope, or building a new one? Had there been an alternative choice, what would have affected your decision?

1.4: Assuming some construction/modification was done, how long did it take to get from initial planning to the start of construction? How many people were on the team, and what kind of expertise did you find you needed?

1.5: Also, how long did it take to get from the start of construction/modification to first light?

1.6: And how long did it take to get from first light to first *robotic* light?

1.7: What kind of testing did you do during integration and/or commissioning to verify that the telescope would work robotically? Was this done in the lab, or only on-site? If only on-site, did this strategy work? (obviously, in a perfect world, of *course* you test in the lab first, but do you consider lab testing to be *essential*?)

2. Technology & installation:

2.1: How did you originally choose your technology, and were there choices you later regretted (and/or changed)?

Were there specific features that made technology decisions for you?

2.2: Has your technology changed over time (mount, OTA, camera, dome, computer OS, software stack)? If so, did this happen by gradual evolution, by planned upgrade/replacement of large parts of the whole system at once, or by a combination? Given what you did, and how it worked out, how are you planning to carry out upgrades in the future?

2.3: What auxiliary equipment did you end up using (weather stations, webcams, rain sensors, etc.) and why? Were there specific features in these devices that guided your choices?

Did you have balance or weight problems on the telescope itself? If so, how did you deal with them.

2.4: How did you mount auxiliary equipment, such as power supplies to cameras, filter wheels, etc. (equipment shelf or custom enclosure/cable-ties + gravity). Did this change over time, or did you Get It Right first time?)

2.5: Similarly, what did you do with all the cables linking the telescope to the outside world? (vanish through the floor?/raised floor?/cable-ramp on floor?/forbidden sector of the dome?) What would you do if you had to do it again?

3. Site and maintenance:

3.1: Did you have problems with heat/cold/humidity/corrosion/wind/lightning/wildlife/earthquakes? Were specific devices affected? What measures did you try to neutralise these problems?

3.2: What on-site support was available? Was this decisive in your choice of site? (or your decision to go ahead?) In practice, what kind of site support have you actually needed? A knowledgeable astronomer? An electrician? Somebody who can close the dome manually?

3.3: What kind of maintenance did you need to do? Which pieces of technology proved least lasting/most vulnerable?

3.4: What planning did you do to deal with technical problems and/or emergencies? (e.g. loss of power, loss of network connection, dome stuck open, etc.) (Think UPS, phone to on-site person, lightning detection, redundant weather sensors....)

3.5: What tools and equipment do you maintain on site? (Or perhaps, what would you advise.... ;-)?)

4. Data and processing?

4.1: Is your data processing done on-site, or back-in-the-lab? Did the quantity of data generated match your predictions? If not, can you explain the factors responsible? (effective hours of darkness/avg. exposure time/bad weather/...)

4.2: What kind of data capacity (in terms of nights) did you install on site? Was this adequate or not, and how has it changed over time.

4.3: What arrangements did you make for retrieving data from the site? What steps did you take to ensure integrity (leave a copy on-site/checksums/...). If you use such steps, have they ever saved you from losing data? (Or vice versa, have you ever lost data for not having them?)

4.4: What arrangements did you make for storing/archiving your data? Did you have a plan for doing this at the beginning, or was it begun only after data started to arrive?

5: Disasters will happen!

5.1: Every big project has at least one disaster story to share.... (can be anonymised to protect the embarrassed!). What happened? How did the problem present itself at first? How did you really figure out what had gone wrong, and how did you fix it?

6: Advice to a novice:

6.1: What advice would you most liked to have received starting out (based on what you now know)? About what and where to build (and whether!)? About technology choices? Installation design? About operation? About the number of people required?

GLORIA Partners

UPM
Universidad Politécnica de Madrid
SPAIN

AUAV
Astronomical Institute, Academy of Sciences of the Czech Republic
CZECH REPUBLIC

CSIC
Consejo Superior de Investigaciones Científicas
SPAIN

INAT & CVUT
Czech Technical University in Prague
CZECH REPUBLIC

IP-ASCR
Institute of Physics of the Academy of Sciences of the Czech Republic
CZECH REPUBLIC

IAC
Instituto de Astrofísica de Canarias
SPAIN

INAF
Istituto Nazionale di Astrofisica
ITALY

SAO
Special Astrophysical Observatory of Russian Academy of Sciences
RUSSIA

UCDNUID
University College Dublin
IRELAND

UC
University of Chile
CHILE

UMA
University of Malaga
SPAIN

UOXF
University of Oxford
UNITED KINGDOM

UNIWARSAW
Uniwersytet Warszawski
POLAND