Using the Web to Plan Telescope Observations

Stephen Wampler
Gemini 8M Telescopes Project
P.O. Box 26732, Tucson, Arizona 85732-6732, USA

Abstract

The new generation of large, ground-based telescopes must operate differently than previous telescopes. The cost of developing and operating these telescopes means that efficient and effective operation has a high priority. While classical observing often means that astronomers work out the details and sequencing of observations while seated in front of the telescopes; the new telescopes require careful planning to reduce costs and to take best advantage of changing viewing conditions.

However, any advance planning tool presents a problem. If it is not convenient and if it does not provide astronomers with the tools to develop strong science programs then the tool does not get used. Further, such a tool should work for astronomers who are working away from the observatory, yet still reflect the conditions at the observatory. While the Internet and World-Wide Web address the problems of off-site support for science planning, traditional Internet-based solutions introduce logistical problems. Similarly, traditional Web-based approaches can quickly dissolve in user frustration over poor network bandwidth and excessive delays.

Fortunately, new Web-based technologies address these limitations. The Gemini 8M Telescopes Project has adopted some of these technologies in the science programming tools for use on Gemini telescopes. These tools are used on- and off-site for both the development and the execution of science programs, and works for both classically performed observations and preplanned, queue-based observations. This paper describes these tools and the underlying technologies.

1 Observing with the Gemini 8M telescopes

Gemini is constructing two 8-meter aperture telescopes. The first is located on Mauna Kea in Hawaii while the second is located on Cerro Pachon in Chile. Together, both telescopes give full sky coverage with diffraction-limited seeing in the near-infrared. Seeing image quality for ground based observing is heavily dependent upon atmospheric conditions, with the best *seeing* conditions occurring approximately 10% of the time. For this reason, science programs are characterized by their seeing requirements. This allows on-site observers to match observations to existing conditions when the observatory is operating using a queue of prepared science programs. This, in turn, presupposes advance preparation of science programs[1].

Science program preparation for the Gemini telescopes is a two-step process[2]. During *Phase 1* proposal preparation, scientists describe the science they would like to perform using a Gemini telescope. Along with other information, a proposal includes information about the instrument configuration, celestrial target selection, exposure times, and required seeing conditions for each observation in the proposal. This proposal then undergoes a review process that determines the suitability and significance of the science that is involved. Proposals that are accepted for execution at a Gemini telescope then enters *Phase 2* preparation, where a formal *science program*[3] is constructed. This science program fully describes the sequence of observations to be carried out by the telescope.

When a proposal is awarded *classical observing* time, it is scheduled to run in a series of blocks under the control of the astronomer or an on-site observer. During classical observing, the astronomer is taking a chance that the seeing conditions are sufficiently good during the scheduled time. If conditions are not good enough, the science program is wasted. Proposals scheduled for *queue observing*, however, are placed into a pool of queue-based programs. Staff astronomers can match seeing conditions to seeing requirements for the observations in this pool to select the best observation to perform. (Other criteria, such as a scientific merit rating, are factors in this selection process.)

2 Tools for program preparation

Effective use of the Gemini telescopes requires that science programs be prepared well in advance of actual use of the telescopes. There are a number of factors that impact this preparation.

- The Gemini partnership consists of six countries, widely distributed around the globe (United States, United Kingdom, Canada, Chile, Argentina, and Brazil).
- Some sites within the partner countries suffer severe performance limitations when using the Internet.
- Sites within the partnership use a variety of different computer systems.
- Gemini resources are limited.

The Internet and the World-Wide Web are natural resources to use in support of proposal and science program preparation. However, experience with bandwidth limitations and network delays has shown that traditional web (typically cgi-script based) are not suitable. Furthermore, this approach relies heavily on active connections to a central site, which is problematic given the locations of the observatory facilities. This leads to a solution where software is distributed from the central site for execution off-site. Traditionally, this means bundling the software into an archive and then making the archive available via ftp. Either the source or a binary distribution can be provided in this manner.

Distributing software has its own problems, particularly in a heterogenous environment of mixed computer systems. Such distributions require large amounts of software support to handle a multiplatform distribution. An additional problem is keeping the distributed software synchronized. Since instrumentation at the observatory changes over time it is important that astronomers use the current version of a tool to ensure compatibility.

2.1 New technologies

Recently, several new technologies have been introduced that provide an infrastructure that address the above issues.

First, the emergence of *Java* as a system independent, high-level programming language has greatly reduced the problem of supporting software across a multiplatform environment. Java's object-oriented design, support for graphical user interfaces, and system independence are all useful programming features.

Second, the support for *push* technology on the Web helps with the problems of distributing and maintaining software in a widely distributed environment. For example, Marimba's Castanet technology allows software to effectively update itself as needed by automatically connecting to the central site to obtain updates. In this way users can be assured of access to the most recent version of a software tool.

The Gemini tools for proposal and program preparation are written in Java and distributed with Castanet. When running, none of the tools require a network connection to the observatory.

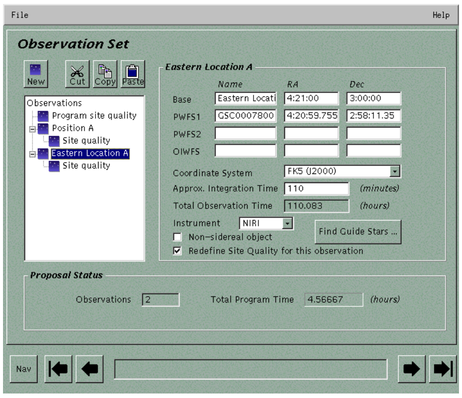
2.2 Phase 1 proposal preparation

The Phase 1 proposal tool[4] is intended to simplify the astronomer's entry of required information and the automatic creation of a draft science program from the proposal. The tool is entirely GUI-based and includes navigational aids as well as consistency checks. Figures 1 and 2 show two of the screens from the tool.

In Figure 1, the astronomer is entering information about the observations that need to be taken as part of the proposal. Since obtaining high-quality data with a Gemini telescope requires appropriate guide stars near each target, selecting Find Guide Stars... results in a catalog search to determine if any guide stars are available for the current target.

Figure 2 shows the navigation window. As entry screens are completed, they are checked off in the navigation window. This allows the astronomer to quickly determine which parts of the proposal need more work.

When the astronomer is satisfied with the proposal, the tools submits it to the screening board, or *Time Allocation Committee*(TAC) for the astronomer's country. Proposals that are accepted by the country's TAC are then submitted to a TAC for the entire partnership of countries. Successful proposals end up at the Gemini observatory. The Phase tool submits the proposal as a rudimentary science program that can then be complete during the Phase 2 process.



Observation Set

Observations
- Program site quality
- Position A
  - Site quality
- Eastern Location A
  - Site quality

Eastern Location A

	Name	*RA*	*Dec*
Base	Eastern Locati	4:21:00	3:00:00
PWFS1	GSC0007800	4:20:59.755	2:58:11.35
PWFS2			
OIWFS			

Coordinate System FK5 (J2000)

Approx. Integration Time 110 *(minutes)*

Total Observation Time 110.083 *(hours)*

Instrument NIRI

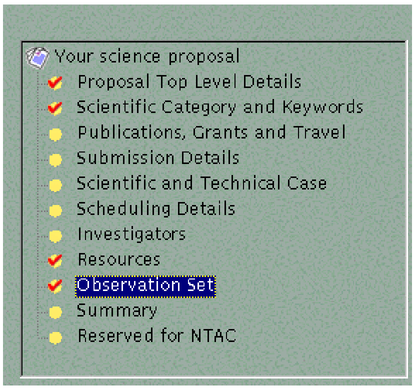
☐ Non-sidereal object

☑ Redefine Site Quality for this observation

Proposal Status

Observations 2 Total Program Time 4.56667 *(hours)*

Figure 1 Phase 1 observation description



Your science proposal
- ✓ Proposal Top Level Details
- ✓ Scientific Category and Keywords
- ● Publications, Grants and Travel
- ● Submission Details
- ● Scientific and Technical Case
- ● Scheduling Details
- ● Investigators
- ✓ Resources
- ✓ Observation Set
- ● Summary
- ● Reserved for NTAC

Figure 2 Phase 1 navigator

2.3 The science program

A science program (Figure 3) describes, in detail, the information needed to perform the requisite science. This program is a hierarchical collection of objects. Basic levels in the hierarchy are the *program*, groups of observations called *folders*, *observations*, *components*, and *iterators*.

The program level collects a potentially large set of observations all related to a single science goal. Programs map one-to-one with proposals. Observations collect information for one or more science frames associated with a given target position in the sky. Observations range from a simple stare at a given RA/Dec to complex mosaic patterns, etc. Observations consist of various components and an (optional) iterator. The components describe the static setup and scheduling constraints of an observation, for instance, the telescope slew target, guide star selection, initial instrument configuration, etc. Iterators describe how to operate on the static components in the observation to produce science data. Example iterators include offset patterns for dithers and mosaics and cycles of instrument filters. Iterators allow sequences of repetitive tasks to be expressed concisely, even if some properties (e.g., offset position or filter selection) vary from exposure to exposure.

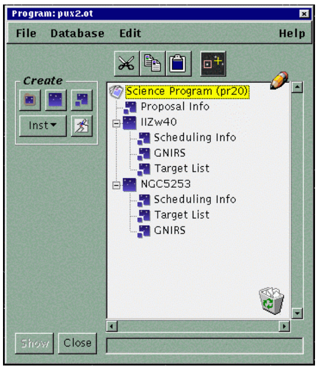
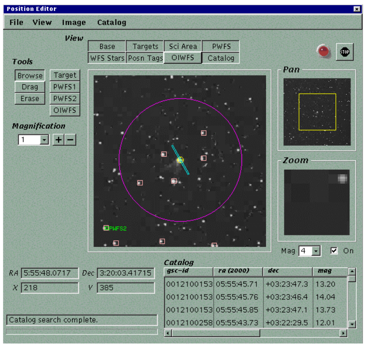


Figure 3 A Science Program (observing tool view)

2.4 The observing tool

Phase 2 preparation is done using the *Observing Tool[5]*. This tool allows the astronomer to provide detailed information about the planned science, including full sequencing of the individual observations involved. Unless the astronomer constrains the observations contained in the science program, individual observations may be executed in any order, at any time. This allows the observatory staff to schedule observations to make the most efficient use of changing conditions. The astronomer may, however, *chain* observations to be executed in a specific sequence and also *group* observations to execute as a single block.

Components within an observation are configured using an appropriate configuration editor - there are different configuration editors for the different components. For example, the editor for the Near Infrared Imager allows the astronomer to select filters, exposure times, etc., while the *target editor* allows the astronomer to associate specific targets and guide stars with a given observation. Because many pieces of configuration information relate to positions on the sky, a *position editor* is available for visually selecting and modifying these items. The position editor (Figure 4) can display an image of a region of the sky and overlay target positions, nearby guide stars, auxiliary probe positions, and science detector position and orientation.



Magnification 1

Mag 4 ☑ On

RA 5:55:48.0717 Dec 3:20:03.41715

X 218 Y 385

Catalog search complete.

Catalog

gsc-id	ra (2000)	dec	mag
0012100153	05:55:45.71	+03:23:47.3	13.20
0012100153	05:55:45.76	+03:23:46.4	14.04
0012100153	05:55:45.85	+03:23:47.1	13.73
0012100258	05:55:43.73	+03:22:29.5	12.01

Figure 4 The position editor

Probes, target positions, and science detectors can then be moved to new positions using the mouse. The sky image is typically loaded from one of a number of Web-based sky survey catalogs, as are the nearby guide stars. Images can also be loaded from local files if desired.

When using the observing tool to complete a science program, the astronomer must first access the observatory and retrieve a copy of the science program. While working, the astronomer may choose to save the program locally or store it back at the observatory. When the science program is ready, it must be stored back at the observatory for scheduling and execution. The Observing Tool uses Sun's *Joe* (*Java Objects Everywhere*, based on CORBA/IIOP technologies) to communicate with the observatory.

One goal of the design of the Observing Tool is that it also function on-site during interactive observing. This provides astronomers with a common tool whether planning observations or interacting directly with the telescope during classical observing. During interactive use, the Observing Tool again uses Joe to interact with the control system. Observations may be submitted for immediate scheduling and execution.

3 Status

Both tools described here are currently in Beta test. So far, both tools have been used on Sun Solaris, Linux, MacOS, Windows NT, and Windows '95 with no modifications to the tools (a shell script used to start the tools required minor modifications to work with the Windows operating systems). Although there are performance concerns on some systems over the use of Java, this will improve as Java matures. Both tools work well when used on laptops, even when removed from the Web. Naturally, some functionality is lost when so isolated, but these operations are quickly restored when reconnected to the Web.

Acknowledgments

The Phase 1 proposal tool was written by Dayle Kotturi. The Observing Tool was written by Shane Walker. Kim Gillies provided significant technical support, including major infrastructure components, to both tools. Principal scientific direction came from Phil Puxley.

References

[1] Wampler, et al, *Software Design Description*, Controls Group, Gemini 8M Telescopes Project, 1994.

[2] Wampler, Gillies, Puxley, and Walker, ``Science planning for the Gemini 8M Telescopes", in *Telescope Control Systems II*, Hilton Lewis, Editor, Proceedings of SPIE Vol. 3112, 246-253, 1997.

[3] Walker et al., *Observing Tool Design Foundations*, Controls Group, Gemini 8M Telescopes Project, 1997.

[4] Kotturi, Dayle, ``Gemini Phase 1 Science Proposal Entry Tool", SPIE Conference proceedings, 1998.

[5] Walker, et al, “Science planning with the Gemini Observing Tool”, SPIE Conference proceedings, 1998.