

The Phase II Language for the Hobby*Eberly Telescope

Niall I. Gaffney

*Hobby*Eberly Telescope, RLM 15.308 University of Texas, Austin, TX 78712, Email: niall@astro.as.utexas.edu*

Mark E. Cornell

McDonald Observatory, RLM 15.308 University of Texas, Austin, TX 78712, Email: cornell@puck.as.utexas.edu

Abstract. We describe our recent software effort to enable queued observations on the Hobby*Eberly Telescope. For the past year, we have worked on development and implementation of a “Phase II” style observation submission and execution system. We have developed a language in which investigators can describe their observations, a simple method for submission, and tools to execute observations. We discuss the special features of this language that were implemented to accommodate some of the unique characteristics of the Hobby*Eberly Telescope.

1. Introduction

The Hobby*Eberly Telescope (HET) (Sebring & Ramsey 1994) is a 9.2 meter optical telescope currently undergoing scientific commissioning in west Texas. The telescope is operated by McDonald Observatory on behalf of the HET consortium¹. Once fully operational, this telescope will operate in a queue scheduled, service observing mode 85% of the time.

What distinguishes the HET from other optical telescopes is its fixed elevation of 55°. The telescope acquires and tracks an object by rotating in azimuth and moving an Arecibo-style tracker mechanism across the focal surface. While this procedure allows the HET to access over 70% of the sky over the course of the year, each object is only observable for approximately 1 hour once or twice a night. We therefore maximize the productivity of this telescope with queue scheduled service observing. To facilitate queue scheduling, we have constructed a scripting language in which investigators can describe their observations in familiar astronomical terms. Using these observing scripts, the HET Operations Staff can build a queue of observations and then execute the observations as required by the investigator.

¹The HET consortium members are The University of Texas at Austin, The Pennsylvania State University, Stanford University, Ludwig-Maximilians-Universität München, and Georg-August-Universität Göttingen.

2. The Phase II Language

Because of the wide variety of computers in use by our investigators, we needed to develop a platform-independent Phase II system. We chose to use a text-based system of keywords and parameters for the language. These keywords are contained within modular “templates”, each of which represent a simple observational concept. These templates are:

- Summary template – Investigators’ contact information and projects’ scientific goals
- Object/Calibrator template – Information about the object (e.g., position, magnitude)
- Instrument template – Setup information for the instrument
- Constraints template – Conditions acceptable for observations (e.g., sky brightness)
- Lamp/Flat/Dark template – Setup information for unique calibrations required for the data
- Plan template – The sequence of actions to be taken to acquire data

The investigator is able to specify details such as which dates and times are acceptable for the observations, what observing conditions are acceptable, how periodic or phase critical observations are to be carried out, and what the relative priority of different parts of the project are to the overall scientific goal. All of this information is specified in terms familiar to an astronomer with concepts similar to what one would tell an experienced observer unfamiliar with the specifics of the project.

Each template can be called many times from within one plan or from multiple plans, saving the investigator from having to make multiple entries for the same object or instrument setup. In addition to this reusability, we have implemented macros to reduce repetitive typing. Recursive macro calls also allow the user to create complex plans with minimal typing. Details of the language are further described in the Phase II manual².

3. Unique Features

Because of the constrained motion of the HET, we needed to develop two unique features for our Phase II language. The first of these is how one requests an observation to be done. One can *do* an observation, which means that the observations are to be executed in the given sequence with no other observations intervening. Alternately one can *schedule* an observation. This allows the HET queue to place that observation at any time in the night that is convenient and allows other observations (possibly from other investigators) to take place

²http://rhea.as.utexas.edu/HET_documentation

<pre> begin_object: SAO10 description: protogalaxy ra: 00:04:45 dec: +22:43:26 equinox: J2000 pa: 142.3 ra_offset: 10.3 dec_offset: -23.4 # magnitude of acquisition object v: 12.4 # flux of object flux 8300 continuum 21.4 rv: 430.3 finding_chart: sao10.fit end_object: </pre>	<pre> begin_plan: SAO10 instrument: lrs_setup1 constraint: very_good class: required # starts on the 18th of November # ±3.5 days utdate_start: 18/11/1997 delta 3 12:00 do-flat: 50 5 do-object: SAO10 800 scaled 2 do-lamp: ThAr 10 5 schedule-dark: 500 5 schedule-calibrator: bluestar 50 fixed 1 # wait 3 ±1 days schedule-wait: 3 delta 1 do-object: SAO10 800 scaled 2 end_plan: </pre>
---	--

Table 1. An example object template (left) and plan (right). Lines preceded by the “#” character are comments.

during the interim. The plan in the right column of Table 1 demonstrates both of these commands. The sequence of “do” commands require that a flat lamp be taken immediately before and a wavelength calibrator be taken immediately after the observations of the object. The first two “schedule” commands allow the HET Operation staff to make those observations at any time in the night. The schedule-wait command tells the observer to wait for a period of time, allowing other observations to take place during the interim.

The second new feature allows the investigator to give either static or dynamic integration times. As the HET tracks an object across its 11 meter primary mirror, the 9.2 meter beam becomes vignetted. This vignetting is a function of the location of the object on the sky, the time the object was acquired, the length of the track, and the azimuth of the telescope. Because the investigator will typically not know when his/her observation will be scheduled, we allow the investigator to either describe an exposure time that will scaled according to this vignetting function and the observing conditions at the start of the exposure, or to statically set the integration time (e.g., if the exposure is time critical). Exposure time scaling is done to conserve the signal-to-noise ratio of the observation based on the vignetting function and estimates of the atmospheric transmission and seeing losses.

4. Plan Submission and Execution

Once plans are created by the investigator, they are e-mailed to the HET Operations Team in Austin. Upon receipt, the plans are automatically validated using a procmail-type³ validation system. Approved plans are then written to disk where, at a fixed time each day, any changes and additions are compiled into the HET Observation relational database. This database is a Starbase database (Roll 1996) to which we have added some new HET-related calculations.

³<http://www.ling.helsinki.fi/eriksso/procmail/mini-faq.html>



Figure 1. The plan execution tool reads in plan files and coordinates the observations with the other subsystems.

The database is transferred over the Internet to the HET site at the beginning of each night. The database will then be used by the resident astronomer to schedule plans throughout the night (Gaffney & Cornell 1997). When a scheduled target becomes observable, the plan is loaded into the plan execution tool (Figure 1). This tool displays the relational database information in an easy-to-digest GUI format. This GUI sends coordinates to the telescope control system and creates macros to be executed on the instrument control systems (ICS) to acquire the data. When the data is written to disk by the ICS, this tool logs the data as being acquired, appends comments to and fixes up the FITS headers to match the standard header format for the HET, moves the data to the correct directory, and queues the data for backup and transfer to Austin.

References

- Gaffney, N. I., & Cornell, M. E. 1997 in ASP Conf. Ser., Vol. 125, *Astronomical Data Analysis Software and Systems VI*, ed. Gareth Hunt & H. E. Payne (San Francisco: ASP), 379
- Roll, J. 1996, in ASP Conf. Ser., Vol. 101, *Astronomical Data Analysis Software and Systems V*, ed. George H. Jacoby & Jeannette Barnes (San Francisco: ASP), 536
- Sebring, T. A., & Ramsey, L. W. 1994, in *Advanced Technology Optical Telescopes V*, SPIE Tech Conf. 2199