Robert C. Byrd Green Bank Telescope
NRAO Green Bank

Dana S. Balser, J.R. Fisher, Anthony H. Minter & Richard M. Prestage

January 11th, 2002

GBT SOFTWARE PROJECT NOTE 18.1

GBT Observe (GO): Initial System Requirements

HTML version Available[1]

Contents

Abstract

This document summarizes the main requirements for GBT Observe (GO), the observer's User Interface for the GBT. In many areas, it simply restates what has already been implemented. The major new topic of the document is to specify how the observing system should be configured prior to starting the scan.

This document was reviewed by the GBT commissioning scientists, together with members of the GBT M & C group and the authors, and accepted on 19th December 2001.

[1] http://www.gb.nrao.edu/GBT/MC/doc/dataproc/GOreq/GOreq/GOreq.html

1 Design Philosophy

The GBT monitor and control system (Ygor) manages many devices consisting of both hardware and software. The system is designed to be general and flexible such that the observer can configure the system in ways which might not have been realized by the observatory staff. The GBT observer interface (GO) should provide an interface specifically tailored for the observer. GO should employ a layered design. Currently, to configure the GBT using Ygor requires many parameters to be set. GO should be designed such that at the top layer these parameters are set with only a limited number of observer input parameters. Sensible defaults for many of the Ygor parameters will have to be defined. This approach requires that a standard set of observing modes be defined which should accommodate over 90% of the observing proposals. However, GO should allow for complete access to the underlying system, Ygor.

GO should have an interactive GUI with both monitor and control panels. There should be a layered approach where the most important parameters are at the top level. GO should also be able to run in batch mode by executing an observing file or table (GO table). There should be software available to aid the observer in generating an observing schedule. For example, given a list of sources, times, and procedures the software would calculate a complete schedule. In principle, a GO table could be generated. The observer should have easy access to various calibrator lists from within GO.

There should be a well defined and tested process to setup and configure the GBT before the actual observing starts. This process should occur in phases. The first phase would be to initialize the GBT. That is, there should be a default configuration for each GBT device and a mechanism to initialize the system from GO. Second, a limited number of GO parameters (keywords) should be specified which will configure the GBT for the current observations. That is, the router switches, the adjustable filters and local oscillators, and the attenuators will be set by GO such that the proper IF power is being sent from the front end to the back end. Third, there should be a testing and tweaking phase where the user has the opportunity to test the system. Additional keywords will probably have to be set. Also, the observer might want to inject a test tone at the front end to check the IF system configuration. Lastly, the observer starts the observing program.

There are several ways to define an observation. That is, how to organize the scheduling and data collection process. Observing procedures should be the standard mechanism to begin an observation. That is, an observation should be defined by an observing procedure. An observing procedure is a function which specifies how to move the telescope and collect data. An observing procedure should consist of one or more scans where a scan is a contiguous period of data collection predefined by a set of Ygor parameters. For example, consider a procedure which consists of a five point map typically used to determine local pointing corrections. The procedure might perform five scans, one centered on the expected location of a pointing source and four at cardinal direction around the center location. Therefore the five point procedure would instruct the telescope to move to these locations and collect data for some specified time. A observing file or GO table may consist of many observations. There should be a number of standard observing procedures available to the user. Moreover, it should be fairly straightforward for the observers to write their own procedures. Also, the parameters which are used by the procedure should be easily accessible in the data reduction package. The observer should be able to configure the system for the next observation while the current one is being executed.

GO should be compact enough such that only one monitor screen can display the relevant information without having to open and close many windows. Multiple instances of GO must be fully synchronized. That is, the remote observer and the operator will necessarily be running GO simultaneously. Actions taken by one user should be reflected by the other. This should include not only the Ygor parameters by GO user-specific parameters as well. The observer should be able to control the telescope remotely.[1]

2 Observer Applications

The GBT observer interface (GO) should contain applications which aid the user in a productive and efficient manner. There should be sensible error checking. Below we list the essential ingredients for GO.

[1]Requirements concerning remote observing and security will be dicussed elsewhere.

2.1 Graphical User Interfaces (GUIs)

GO should have an interactive GUI with both monitor and control panels. The GUIs should behave similar to those developed in most standard software packages. A layered approach should be employed where the most important parameters are displayed on main panels. There should be three main panels: a control panel, a monitor panel, and a status panel.

2.1.1 Control Panel

The main control panel should include information from various devices which plan to be utilized in the observing. For example, the front-end receiver, the back-end configuration, any relevant frequency information, the observing and switching modes to be utilized, etc. There should also be information about the source, its coordinates, velocity information for spectroscopy, pulse period for pulsars, etc. Generic information about the observing program should be listed (e.g., the observer and project ID). Information about the procedure, which defines the observation, should be listed along with the relevant procedure parameters.

Additional details should be provided from underlying panels with the philosophy that most of the relevant information is on the main panel. Thus, there needs to be a balance. Too much information on the main panel make it difficult to absorb, while too little requires the observer to bring up additional panels.

2.1.2 Monitor Panel

The main monitor panel should focus on the antenna. This should be done graphically where telescope position and its scheduled movement for the current scan is indicated. Also, the user should be able to display the positions of sources from a catalog which can be specified from an ASCII file. Important information from other devices should also be displayed on the main monitor panel. There should be information about the weather, the tipper for high frequency observations, the total power levels, and any relevant frequency information.

Underlying GUIs should contain information which the observer would not regularly wish to monitor. It should display graphically the available frequency band in RF units from the front-end to the back-end. This might only be viewed once during the initial setup. Also display the power levels from the front-end to the back-end at critical places.

2.1.3 Status Panel

The main status panel should reflect the current status of the system. This panel need not be very sophisticated. A “christmas-tree-light” approach is desirable. The main purpose of the status panel would be to let the user know that all devices are working properly. Additional information should exit in underlying panels. However, it is the job of the observatory staff to fix any problems with the system and thus the tools to perform repairs need not be located within GO.

2.2 Observing Files

GO should also be able to run in batch mode by executing an observing file (GO table). These should be simple ASCII files which can be manipulated with any standard text editor. Software to check the sanity of these files should be available. At a minimum the syntax of the GO table should be verified. Because of the complexity of the GBT there are hundreds of parameters which are required to configure the system. A limited number of well documented GO parameters (keywords) should be defined that are tailored for the astronomer which can be used to configure the system. These keywords need to be mapped onto the lower-level monitor & control (Ygor) parameters which are device specific. However, GO should be able to set each of the Ygor parameters as well with the intention that this would not be normal operation.

2.3 Observing Schedules

Software which aids the observer in planning their observing schedule should be developed. It should be possible to import an ASCII file containing a source list. In principle this could be the same format as that used for the GO table. The software should allow the user to plan the observing schedule and at least generate scheduling reports. That is, information about when each observation begins and ends, where the source is located on the sky, slew times, etc. An interactive mode would also be useful. It should be possible to generate a GO table.

2.4 Calibration Lists

A calibrator lists should be available to the user within GO. There should at least be a continuum, spectral line, and pulsar list of calibrators.

3 Observing Process

How should GO be used? Ultimately this should be left to the observer. However, because the GBT has many possible modes there should be software in place to aid the user based on general experience using this system. Because there are hundreds of Ygor parameters there should be some mechanism to set these parameters to some sensible default settings. For example, under most circumstances the observer will want to use refraction. There may be instances where this feature has been deactivated, e.g., during maintenance. Therefore there should be a way to set all parameters to some default ("normal") setting. At the next stage, once the Ygor parameters are set to their default values, there are still device parameters which must change. This is a result of the variety of front-end receivers and back-ends which are available with the GBT. Therefore there should be software within GO which configures most of the system based on a limited number of keywords. Third, there should be a phase where various tweaking and testing can be done to verify that the system is ready for observations. Lastly, the observations begin. Below we discuss these stages in more detail.

3.1 Initialization

There should be a default configuration for each GBT device which can be set from GO. This is necessary since during maintenance, for example, some Ygor parameters may be set to non-standard values. It should be possible to initialize the system from a GO table. This could be done by including a Glish script from within a GO table or by running a GO table itself. The observer should *not* have to learn Glish to configure the system and any initialization scripts written in Glish are expected to be written and maintained by the observatory staff. These default settings must be well documented. Also, it is desirable that they do not change often.

3.2 Configuration

There should be a limited number ($\leq$ 12) of GO keywords which act like meta-parameters and will configure the system for an observing run. That is, the router switches, the adjustable filters and local oscillators, and the attenuators will be set by GO such that the proper IF power is being sent from the front end to the back end. GO will have to choose several default settings when the solutions are degenerate. For example, if there are two paths that a particular IF signal can get to the specified back-end GO should choose one as the default. In practice more than one observing mode might be configured before the observations begin. For example, the observing run might consist of some pointing observations using the digital continuum receiver and then a spectral line experiment using the correlator spectrometer. Below is a list of the GO meta-keywords.

1. receiver: The front end receiver to be used. Only one receiver can be used at a time (Table 1). The keyword values are given in terms of wavelength. For example, to select the L band receiver sc.receiver = '21cm'.

Table 1: **receiver**

Band	Freq. Range (GHz)	Keyword Value
PF1 (1)	.290–.395	90cm
PF1 (2)	.385–.520	75cm
PF1 (3)	.510–.690	50cm
PF1 (4)	.680–.920	35cm
PF2	.910–1.23	30cm
L band	1.15–1.73	21cm
S band	1.73–2.60	15cm
C band	3.95–5.85	6cm
X band	8.00–10.1	3cm
Ku band	12.0–15.4	2cm
K band (L)	18.0–22.4	15mm
K band (U)	22.0–26.5	12mm
Q band	40.0–50.0	7mm

Table 2: **obs_mode**

Back End	Mode	Keyword Value
DCR	Continuum	dcr
SPM	Spectral Line	spm_line
SPM	Pulsar Timing	spm_pulsar_timing
SPM	Pulsar Search	spm_pulsar_search
SP	Spectral Line	sp_line
SP	Pular Timing	sp_pulsar_timing
SP	Pulsar Dedispersion	sp_dedispersion
BCPM	Pulsar Timing	bcpm_pulsar_timing
BCPM	Pulsar Search	bcpm_pulsar_search
BCPM	Pulsar Monitoring	bcpm_pulsar_mon
BCPM	Pulsar Voltage Monitoring	bcpm_pulsar_vol_mon
VLBA	VLBA	vlba

2. obs_mode: The observing mode which specifies the back end and type. Table 2 lists the different modes. Notice that only one back-end can be selected at a given instance. A mode could be included which used multiple back-ends, however. The possible back-ends are digital continuum receiver (DCR), correlator spectrometer (SPM), spectral processor (SP), Berkeley-Caltech pulsar machine (BCPM), and VLBA.

3. switch_mode: During an integration or scan it is common for the state of several parameters to change on times scales which are small compared to the integration time or scan time. For example, calibration of the intensity scale is usually determined by injecting noise into the system ON and OFF. Also to remove instabilities in the system a reference spectrum is usually determined for spectroscopy. Any "switching" that employs the primary or secondary optics should be placed into separate scans. Of course this need not be the case and might not be feasible if the scan duration becomes small. This will probably not be the case with the primary since the overhead would be too large but may indeed be significant for the secondary. Experience with the system will be required.

The following modes discussed below should be defined: 'tp', 'tp_nocal', fsw_01', 'fsw_12', 'fsw_0102', 'psw_01', 'bsw_01', 'user'. The *obs_mode* keyword will decide what back-end will control the switching. In all cases except for total power it is assumed that the noise diode will be fired ON and OFF. If for some reason the user does not want one of the standard modes they need to explicitly define the switching mode with the option 'user'.

(i) Total Power (tp): In this mode there is no reference signal and the Sig/Ref state is always SIG (Table 3). When

Table 3: **switch_mode** (tp)

Phase	Time	Cal	Sig/Ref
1	0.0	OFF	SIG
2	0.5	ON	SIG

Table 4: **switch_mode** (tp_nocal)

Phase	Time	Cal	Sig/Ref
1	0.0	OFF	SIG

this mode is selected the user typically is employing total power position switching where separate integrations or scans are taken both ON and OFF the astronomical source.

(ii) Total Power without CAL (tp_nocal): This is the same as tp except that the cal is not fired (Table 4). This is necessary if the observer decides to fire the cal between scans.

(iii) Frequency Switching (fsw): The reference spectrum is determined by switching between the desired frequency and one or two reference frequencies. Three fsw modes should be specified and are shown in Tables 6, 7, and 8. ν_0 is the signal frequency offset (typically zero), ν_1 is the first reference frequency offset, and ν_2 is the second reference frequency offset. The first fsw mode (fsw_01) is a simple frequency switching where only one offset is used. The second mode (fsw_12) is a symmetric switching which uses two reference offsets. The third mode (fsw_0102) is similar to the first mode except two reference frequency offsets are used, presumably to remove a first-order, frequency dependent gain term in the front end passband. Typically $\nu_1 = -\nu_2$ but this need not be the case.

(iii) Polarization Switching (psw): Switching between different orthogonal polarizations can help minimize instrumental effects such as receiver gain drift. A switch is used in the front end which toggles the orthogonal polarizations between a given IF output. In Figure 8 XL refers to either X-linear or LCP and YR to either Y-linear or RCP.

(iv) Beam Switching (bsw): For frequencies above 10 GHz there are two or more beams continuously on the sky. Therefore switching between beams is possible by toggling an electronic switch between two states and no optics are involved. An example is given in Figure 9 where only two beams are considered (1 and 2). Note that the Q band system has four beams and additional bsw modes will have to be added.

There are many more possible switching modes which can be left up to the user to define explicitly. If one of these other modes becomes popular it can be added to the list of standard modes.

4. rest_freq: An array of rest frequencies specifying the rest frequency of the center of the band for each channel. For example, if the observer wants to observe 4 transitions centered at 8400, 8500, 8600, and 8700 MHz each with a bandwidth of 50 MHz then *rest_freq* = [8400, 8500, 8600, 8700]. Notice that the number of IF's is implicit in the *rest_freq* array size.

5. bandwidth: An array of the bandwidths specifying the desired bandwidth of each channel. In the example above *bandwidth* = [50, 50, 50, 50]. Table 10 lists the possible back-end bandwidths available.

Table 5: **switch_mode** (fsw_01)

Phase	Time	Cal	Sig/Ref	ν offset
1	0.0	OFF	SIG	ν_0
2	0.25	ON	SIG	ν_0
3	0.5	OFF	REF	ν_1
4	0.75	ON	REF	ν_1

Table 6: **switch_mode** (fsw_12)

Phase	Time	Cal	Sig/Ref	ν offset
1	0.0	OFF	SIG	ν_1
2	0.25	ON	SIG	ν_1
3	0.5	OFF	REF	ν_2
4	0.75	ON	REF	ν_2

Table 7: **switch_mode** (fsw_0102)

Phase	Time	Cal	Sig/Ref	ν offset
1	0.0	OFF	SIG	ν_0
2	0.125	ON	SIG	ν_0
3	0.25	OFF	REF	ν_1
4	0.375	ON	REF	ν_1
5	0.50	OFF	SIG	ν_0
6	0.625	ON	SIG	ν_0
7	0.75	OFF	REF	ν_2
8	0.875	ON	REF	ν_2

Table 8: **switch_mode** (psw_01)

Phase	Time	Cal	Sig/Ref	Pol
1	0.0	OFF	SIG	XL
2	0.25	ON	SIG	XL
3	0.5	OFF	REF	YR
4	0.75	ON	REF	YR

Table 9: **switch_mode** (bsw_01)

Phase	Time	Cal	Sig/Ref	Beam
1	0.0	OFF	SIG	1
2	0.25	ON	SIG	1
3	0.5	OFF	REF	2
4	0.75	ON	REF	2

Table 10: GBT Back-end Bandwidths

Back-end	Bandwidths (MHz)
DCR	Variable
SPM	12.5, 50, 200, 800
SP	0.078, 0.156, 0.312, 0.625, 1.25, 2.5, 5.0, 10.0, 20.0, 40.0
BCPM	—

3.3 Testing

Once the basic connectivity is established the observer may want to check that the system is indeed configured properly. Are the power levels reasonable? Are the frequency conversions correct? A testtone could be injected into the front-end receiver and detected with the back-end for analysis. The observer may wish to test the setup using an astronomical source. There should be some basic tools for such an analysis. This should be an interactive phase which could be performed by the observer or the operator. The observer may feel that the system is robust and the observing setup reliable enough that this phase can be skipped. As experience is gained with the telescope more automatic testing could be implemented.

3.4 Observing

Lastly, the observations are initiated. This might be via the GO GUI for a single observation or a GO table for one or more observations.