

Policy on Issues Affecting the Development of Graphical User Interfaces for the GBT

M. M. McKinnon

November 20, 1997

1 Introduction.

Users of the GBT will interact with the telescope through graphical user interfaces (GUIs). The primary users are operators, observers, and engineers/technicians, and each user will have different expectations and requirements for the GUIs. The pitfalls associated with developing a single GUI for all users are obvious: (1) the GUI design may be compromised such that the GUI does not provide the basic functionality that is required by a particular user and (2) the task of designing and developing such a GUI can be extremely difficult, if not impossible, for the programmer. To avoid these pitfalls, GUIs must be developed for each user, and the function of each GUI must be clearly defined. Additionally, the GUIs must be defined such that duplication of functionality between them is held to a minimum. Duplication of functionality can equate to the unnecessary duplication of effort on the part of our software development group. Furthermore, we must all understand our overall objectives, as well as the individual efforts and responsibilities of our colleagues, to produce efficient, working interfaces to the telescope in a timely manner.

The purposes of this memorandum are to state the current objective of developing GUI software, to define the function of the GUIs, and to establish guidelines for software development.

2 Economy of Software Development Effort.

The Monitor and Control Group has provided very extensive access to the monitor and control data that are produced by a number of devices on the

telescope. In the spirit of object-oriented programming, a natural temptation exists to develop a simple interface for every single device, and then build yet another layer upon this foundation for more sophisticated monitor and control schemes. Unfortunately we have neither the manpower nor the time to pursue such a systematic approach. Henceforth, the immediate objective of our software development efforts is to provide the basic functionality that is required *by the user* to commission the telescope. We do this by first building the summary screens that are required to carry out an observation. Anyone with experience in radio astronomy operations and observations will know the basic requirements for these screens without prior knowledge of the particular details of the GBT monitor and control system. Once these summary screens are built and functioning, the intermediate and lower-level screens can be developed on a priority basis. We should not spend much time refining these screens initially because they will be optimized as we gain experience with the telescope. Although our current emphasis of software development will be focused upon the commissioning of the telescope, our software development priorities will shift to other areas, such as providing additional functionality to the GUIs, in the GBT's operational phases.

We should also be wary of developing too many screens. An overabundance of screens creates the problem of information overload for the telescope operator. Furthermore, if we adopt the strategy of producing a screen for every device now, it is almost certain that many of these screens will rarely, if ever, be used in the near future. This particular strategy is inefficient because it is a poor use of the limited manpower available in the software development group. Although a large number of monitor points exists on the GBT, we must be able to hold the total number of screens to a manageable level. Perhaps we should keep in mind the example of the VLBA in determining what is a reasonable number of screens. The VLBA, which consists of 10 antennas with at least seven receivers per antenna, has faced a high level of complexity in monitoring a large number of data points. After optimization over the operational lifetime of the VLBA, the telescope's GUIs consist of 65 screens for the antennas, 15 correlator screens, and 20 operator screens. Therefore, the total number of observer, operator, and engineering interfaces for the VLBA does not exceed 100.

3 Definitions and Guidelines.

The following list gives definitions of GUIs and sets guidelines for software development. There are three categories of interfaces - for observers, operators, and engineers. Each interface is designed for one particular user (*e.g.* the telescope operator is the primary user of the operator interface). However, each interface can and will be used by other users.

1. The GBT will be engaged in making astronomical observations the vast majority of the time. During this time, operators will be monitoring the health of the telescope, and observers will be issuing telescope commands for a particular type of observation. Therefore, the operator interface should be optimized for monitoring the telescope, and the observer interface should be optimized for controlling the telescope. For a comparatively small fraction of the time, engineers and technicians will be testing telescope subsystems. Many of these tests may be conducted concurrently with astronomical observations. The three GUIs required to fulfill these basic needs at the GBT are listed and defined below in order of importance and priority of development.
 - (a) The observer interface is a GUI that allows an observer, through the telescope operator, to control the telescope and its subsystems for the purpose of carrying out astronomical observations.
 - (b) The operator interface is a GUI that allows the telescope operator to principally monitor the telescope and its subsystems.
 - (c) The engineer interface is a GUI that allows a technician or engineer to control and monitor subsystems and low-level devices on the telescope for the purpose of equipment testing. Some monitoring of engineering data can be done with conventional test equipment, with a copy of the operator interface, or with the data display tools that are available within AIPS++ (see item 8).

The definition of the operator interface is not intended to deny control of the telescope to the operator. The intent of the definition is to emphasize that the primary function of the operator interface is one of monitoring. The operator may control the telescope and its subsystems through the observer interface and the engineer interface. The operator interface may also contain control capability for motions on

and of the telescope (see item 2); however, where the control of telescope motion will eventually reside (*e.g.* operator interface or observer interface) is more a matter of implementation than policy and will not be discussed in this memorandum.

2. For reasons of safety and in keeping with current management policy, control of the telescope devices which involve mechanical motion on the telescope, such as the azimuth and elevation drives, prime focus boom, receiver turret, surface actuators, and subreflector, will be restricted to the telescope operator.
3. Commands will be issued to the telescope and its subsystems in a manner which maximizes flexibility for the observer but is also consistent with the safe and efficient operation of the telescope. Consequently, two general types of observing modes will be allowed on the telescope: sequential and interactive. In sequential mode, an observer submits an observe file to the telescope operator, and the operator arranges for the execution of the commands contained in the file. In interactive mode, the telescope operator issues commands directly to the devices which control the motions of the telescope, and the observer controls the remaining devices whose parameters can be refined for the observation. For example, the observer will be able to stop a data integration, alter relevant parameters such as backend bandwidth, LO synthesizer setting, and receiver polarization, and then restart the data integration while the telescope is tracking a radio source.
4. Only one user can control a particular device on the telescope. However, multiple users may control separate devices at the same time. In all cases the observer, through the telescope operator, has priority control of the telescope and its subsystems. Control of the devices which are not required for an observation can be released by the telescope operator and given to other users as selected by the operator.
5. The monitoring of telescope hardware by users will not be restricted in general. However, the software development group may want to limit the total number of users that can access a single device at any given time.
6. With the exception of NRAO staff who need access to the astronomical data for purposes of quality control, trouble-shooting, and data

backup, access to the astronomical data will be restricted to the observers who recorded the data.

7. With the exception of the GUIs that have already been developed within the Electronics and Metrology Groups, all GUIs will execute on a Unix workstation and display using X windows.
8. The graphical data displays used to monitor astronomical and engineering data will be the basic tools provided within AIPS++ (*e.g.* aipsview and gbtlogview). We need to be careful to avoid memory overload from the AIPS++ system, as well as the GBT GUIs, on machines needed primarily for telescope control and monitoring.
9. We must be very cautious about allocating money and personnel resources to things that are not absolutely necessary for the commissioning of the telescope or that won't get used much over the lifetime of the telescope.
10. The creation of the operator interface will be the responsibility of the software development group, not telescope operations. However, telescope operations must be heavily involved in defining the interface.