Two spectrograph control displays for the W.M. Keck Telescope

Al Conrad, Bob Kibrick (Lick), John Cromer (CIT)

W. M. Keck Observatory
PO Box 220, Kamuela, Hawaii 96743

ABSTRACT

We analyze the look and feel of each display and discuss its software design. Developed using the Motif Toolkit, each display runs on an X windows work station. In the area of look and feel, we discuss the pros and cons of graphically depicting the instrument light path. In the area of software design, we discuss Motif and Xlib programming techniques.

2. INTRODUCTION

We discuss the two spectrograph control displays depicted in figures 1 and 2. Each of these displays controls optical elements within the spectrograph. We will not discuss here the tools for controlling exposures and displaying images.

In the next section we discuss the advantages of depicting graphically the spectrograph light path. In section four we discuss our philosophy regarding user interaction. In section five we compare two approaches for working within limited screen space. Section six describes the stand alone simulation mode used for training and debugging. Finally, in section seven we discuss the development tools used to create the two displays.

3. WHY A GRAPHICAL LIGHT PATH?

A graphical depiction of the spectrograph light path provides a framework on which the control and status of the optical elements can be laid out in logical groupings. Wordy labels are often avoided. (In fact, as can be seen in figure 1, a status display's meaning is often sufficiently clear from its position and units and no label at all is required.) The resultant saving in screen real estate is used to display more information at the top level.

The graphical light path facilitates troubleshooting sessions and discussion of the instrument's performance. Trouble shooting one of the instruments with a traditional, menu driven display involves paging through tabular data and then mapping the filter settings, focus positions, etc., onto a hard copy schematic. Although the hard copy schematic is usually available for a few weeks following commissioning, it is soon buried in the sea of documentation at the summit, and each operator must map the filter settings, focus positions, etc., onto his or her own personal memory of the instrument's configuration.

0-8194-1493-X/94/$6.00

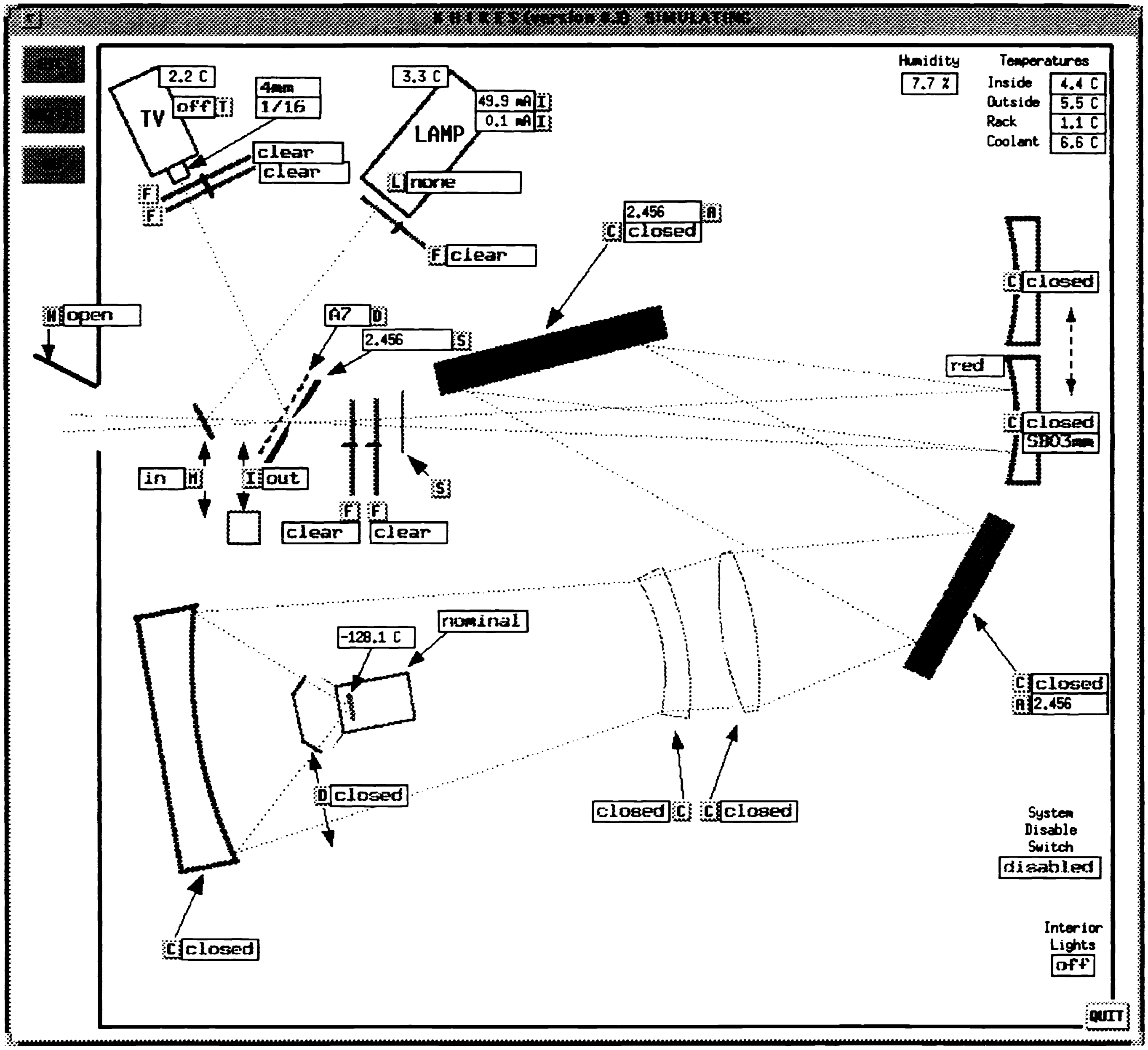


Figure 1. The high resolution echelle spectrograph display: xhires.

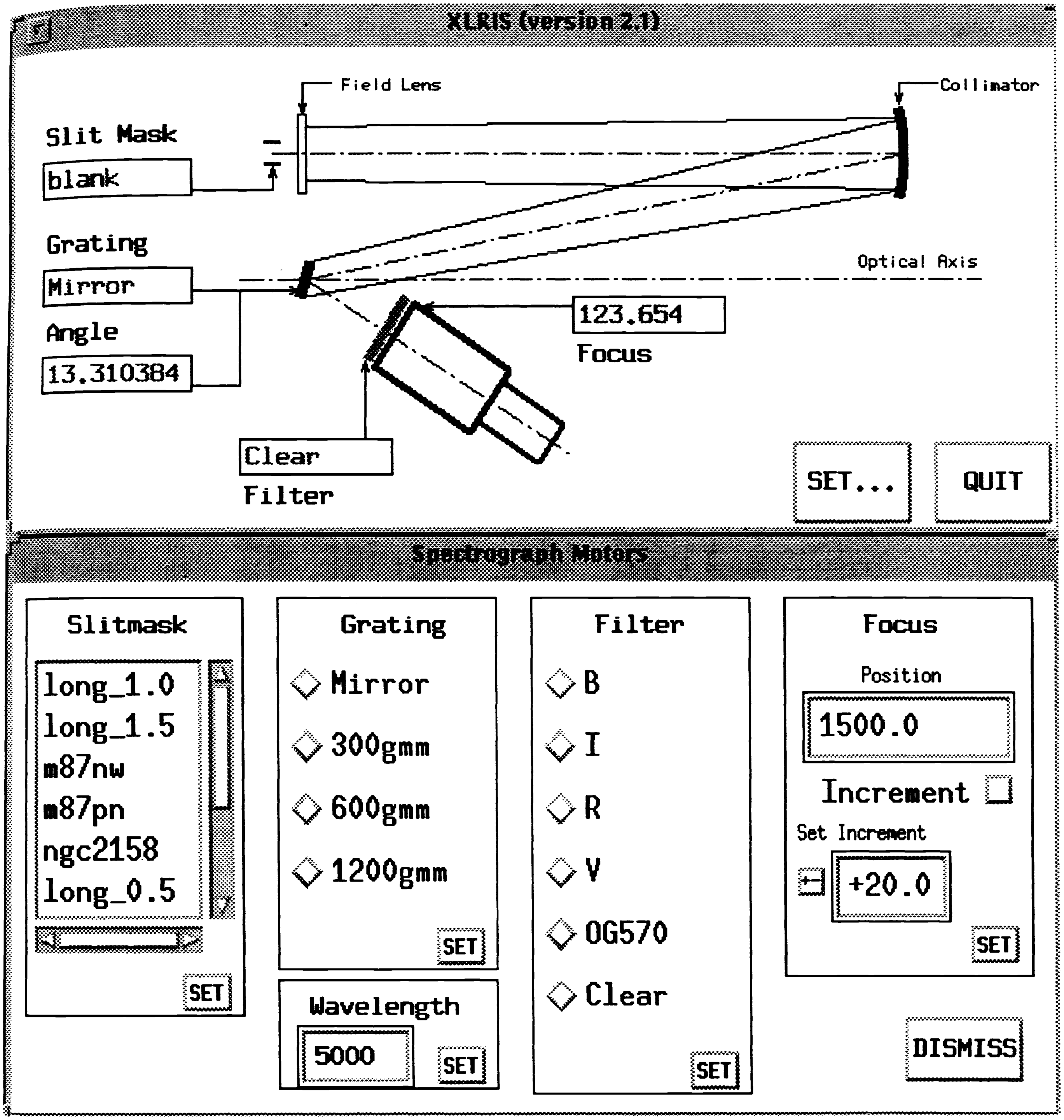
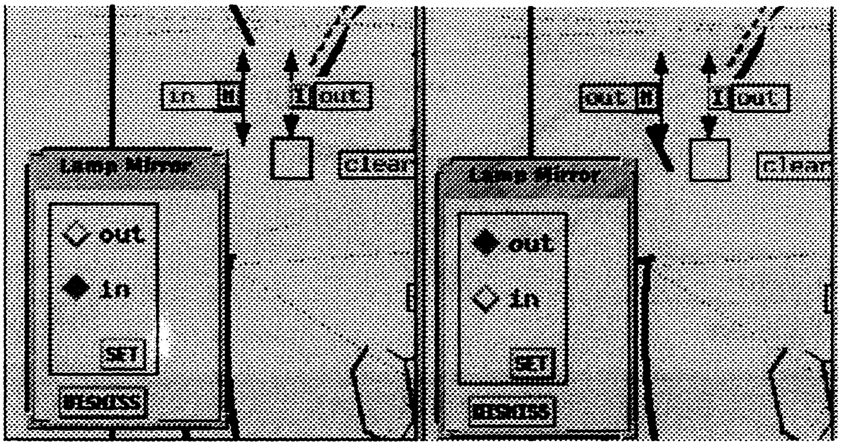


Figure 2. The low resolution imaging spectrograph control display: xlris.

With the *xhires* and *xlris* displays (pronounced *ex-high-rez* and *ex-low-rez*, respectively), collaborating observers often engage in "point-and-talk" sessions during which they share a common view of the instrument's configuration. Because multiple, cooperating copies of the spectrograph display can be brought up on the summit and at remote sites, these collaborative sessions often span large geographical distances.

The graphical light path allows us to display simply and unambiguously the position of optical elements which can be moved into and out of the light path. For example, one cannot confuse whether the *xhires* lamp mirror status "in" means "in" its stow position or "in" the light path:



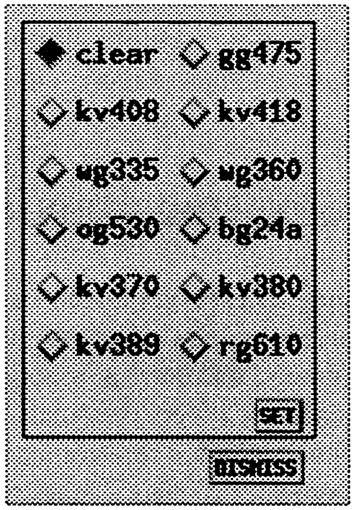
Moreover, the appearance of an optical element blocking the light path often alerts the observer to an error in the instrument set up and thus avoids wasted exposures.

4. USER INTERACTION

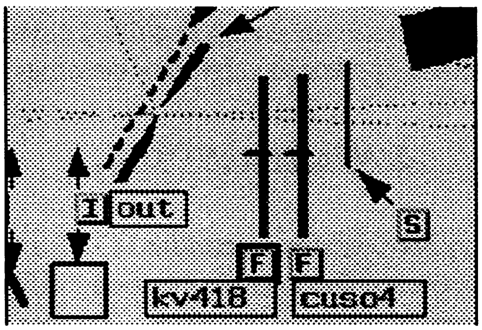
A single click of the left mouse button controls every operation on both *xlris* and *xhires*. We have found that pull right menus, drag-and-drop, and similar mechanisms which require the user to hold down mouse buttons, to move the mouse while holding down, and then to release the button with the cursor hovering above a selected item, are error prone and inappropriate for motor control displays. (Especially to an oxygen-starved observer at 14000 feet.) These mechanisms (pull-right, drag-and-drop, etc.) provide more dimensions on the input side of a graphical user interface and are useful in word processors, spread sheets, and similar tools in which input errors are less disastrous.

Consistency is another important attribute for avoiding input errors. Here we mean consistency across a display or a set of related displays, not adherence to a set of standards intended for the designers of spread sheets and word processors. On both the *xlris* and *xhires* displays, a motor move is always initiated by pushing a button labelled "set". The "set" button for each motor appears on a subordinate pop up window, so the user knows that no operation initiated on the main display will alter the current set up.

We decouple the position display for a given optical element from the control mechanism for that optical element. For example, when the *xhires* filter selection pop up is brought up:



the portion of the display that indicates the currently selected filter:



will not update until the encoder for that filter wheel indicates the new filter is in position. Because hardware state changes drive the display, *xhires* will also display a correct filter position even if someone standing inside the spectrograph moves the filter manually.

5. REAL ESTATE

The complexity of the instrument determines our method for coping with limited screen real estate. The low resolution imaging spectrograph has relatively few optical elements, so we have grouped them on a single pop up window (see figure 2). The high resolution echelle spectrograph has many optical elements, so we give each its own pop up.

Although observers commonly wish to change only a few optical elements, they occasionally wish simply to set up the entire instrument to some predefined state. Both spectrographs use the Keck keyword layer.[1] As a result, the instruments can be set up by extracting the keywords from the image headers of previously acquired images. Alternatively, a set up file can be written from the current state of the instrument from *xhires*. A set up file for the entire instrument can be saved, or just the keywords for certain specific instrument areas, for example, slit area, guider area, etc. Set up files are written using the "write" button, and read in and applied using the "go" button (see figure 1).

6. STAND ALONE SIMULATION MODE

Observers may bring up the *xhires* display in a self contained simulation mode. On earlier user interface displays we implemented external simulators which provided alternative versions of the low level tasks that control the hardware. Although this latter approach gives a more realistic environment for debugging, we prefer the former approach for several reasons. Most importantly, a stand alone simulation mode allows us to export the tool easily to remote sites for training and planning. In particular, observers can use the tool to write set up files as described in the previous section.

The second advantage of a self contained simulation mode is more subtle. We find that a simulation mode which is built into a constantly changing tool is more likely to stay current with new features of the instrument. Typically, programmers use hardware simulators extensively during commissioning, but the simulators grow obsolete as the application software becomes more stable.

7. DEVELOPMENT TOOLS

Both spectrograph control displays are written in standard C with calls to the Motif widget library. During the past few years there has been enormous effort among software vendors to produce more sophisticated methods for building graphical user interfaces. To date, these tools have been targeted toward developers with little or no programming experience. Many of these tools impose cumbersome programming environments or weak, unfamiliar languages. Constrained within these "meta-environments", experienced programmers cannot apply tried and true C programming techniques.

More recently, there has been a welcome shift away from imposing inferior programming paradigms and toward simple products that aid experienced C programmers. As an example, widgets developed for some meta-environments can now be purchased separately within individual subroutine libraries.

Hopefully we will soon see similar steps taken with the drawing programs that are, to date, bundled into meta-environments. In particular, a simple drawing tool which produces statically initialized Xlib structures would be a welcome aid.

8. ACKNOWLEDGMENTS

We thank Steve Vogt and Judy Cohen for providing the specifications for *xhires* and *xlris* (respectively). We also thank Dean Tucker for developing the high resolution echelle spectrograph motor control software. Finally, we thank Julie Barreto for proof reading early drafts of the document.

The W.M. Keck Observatory is operated as a scientific partnership between the California Institute of Technology and the University of California. The observatory was made possible by the generous financial support of the W.M. Keck Foundation.

9. REFERENCES

1. A. R. Conrad and W. F. Lupton, "The Keck Keyword Layer", *Astronomical Data Analysis Software and Systems II, ASP Conference Series*, Vol 52, 1993.