

Considerations in the Design of Graphical User Interfaces for Macromolecular Crystallography

Robert M. Sweet and John. M. Skinner

Biology Department, Brookhaven National Laboratory, Upton, NY 11973, USA

SWEET@BNL.GOV AND SKINNER@BNL.GOV

Abstract

Increases in the number of x-ray beamlines at synchrotron sources and in the number of users of those beamlines require new approaches to development of software. We outline fundamental principles in the art of graphical user interface (GUI) construction and demonstrate these with interfaces we have built to control beamline X12-C at Brookhaven's National Synchrotron Light Source.

1 Introduction -- The Drive for Major Computing Hardware and Software:

X-ray beams are getting hotter and the equipment to use them is becoming more complicated. However, potential users are becoming both more numerous and less experienced. Visitors to synchrotron beam lines are often molecular biologists with the skill and luck to have crystallized a "gene product" that interests them. They (and the structural biological research community) will benefit if they are able to make their measurements efficiently and with a high probability of success. Experienced users should have the full power of modern software and instruments immediately at their fingertips. One needs computing power for data gathering and reduction, and for experimental control: to handle the users needs, providing flexibility when it can be used, but without their needing to know too much.

What are the requirements for computing power?

- Instrument control -- Beamline and data-collection apparatus.
- Data Gathering
 - Data rates will be in the range of 20 MB/sec.
 - Individual data sets will be in the 2-20 GB range.
 - Will need to compromise between price and performance.
- Data reduction
 - One can never have too much computing power.
 - Cheap SMP machines will likely keep up with the load.
- Experimental control -- let's look at *The Grid of Control*. One can define "function" both in terms of what the user wants to do (across the top) and what the software must be like (down the side).

THE GRID OF CONTROL!

FUNCTION	-->	BEAMLINE CONTROL	DATA ACQUISITION	DATA REDUCTION
INTERFACE TO USER		Check status Fix problems Choose wave-length	Check crystal Choose strategy Take the data	Check the start Monitor progress Check the result

CENTRAL CONTROL	Alignment Monitoring Monochromator	Setup (etc!) Iterate Monitor	Generic data reduction
DRIVERS/ DEVICES	Motors Counters	Goniometer Shutter Detector FAST Disk	Disk Network

However, sometimes things are difficult for a user to understand. One needs a graphical interface to help understand what it all means, and good documentation to describe it all. This requires a reasonably powerful front-end computer and carefully constructed software.

Objectives for Construction of New Software

- Assure first the quality and then the quantity of the data that are produced.
- The experimenter must be able to control the measurements and to assess the merit of the results in ways he can understand.
- The experimenter must be helped to do things he doesn't think of, and should be protected from his own errors.

2. Techniques for Graphical User Interface (GUI) Construction:

Many of these techniques come from our own personal experience. New ideas, and articulation of our own experience come also from Brad Blumenthal (Univ. Ill., Chicago) in a lecture from a Workshop on Graphical User Interfaces for Synchrotron Protein Crystallography held at Brookhaven National Laboratory during 19-21 March 1995.

Consistency is the root of GUI design. The Pflugrath "rental car" example defines it: one can sit in any rental car and find the controls. We'd like you to be able to sit at any beamline and find the controls too. In the software, one should use "affordances," devices that have an obvious function. (An example from real life is door handles -- a vertical rod begs to be pulled, a horizontal one to be pushed.) The interface should provide feedback -- colors change or numbers move when something is happening.

An important criterion is to know the "community of practice." This means, the software designer needs to know what the potential users are used to seeing and what they *expect to see* in the software. Things like the "standard" Motif window, recognizable prompts (Files..., Parameters , etc.), and parameters defined in customary dimensions (Ångströms, not nanometers or kilovolts) give the user confidence, and he knows just what to do. Make thoughtful use of the distinction between ritual and habit: the thoughtful repetition of the unnecessary vs. the thoughtless repetition of the necessary. One needs to think through the entire "interface" between the user and the experiment. The balance between hardware, software, and things like pencil and paper needs to be made carefully and flexibly. Look to provide redundancy -- there's more than one way to skin a cat.

In the context of beamline-control software, one should design for the handicapped. Think of the problem faced by weapons designers. Their users are deaf, blind, mobility impaired, emotionally disturbed. In our case, the users suffer sleep deprivation and a noisy and badly lighted environment -- they lose both short-term and long-term memory, become inaccurate and irritable. In short, people running experiments at synchrotrons have Alzheimer's disease (Blumenthal).

Be fearless about trying something and then throwing it away. Prototype in some easy medium. If there's not a GUI builder to help you, prototype with paper or on a chalk board. Very early on, the GUI must be tried on a *naive* user. First try it on friends *in vitro*. Test negative things -- try to break it. Only when it passes all tests in a friendly environment should it be used in anger, on folks who want actually to *use* it rather than to *play* with it. Then only after it passes all tests here should it be used *in vivo* -- after 4 days on the beamline, testing it on a certified idiot.

The most important rule in evaluation is, "Don't use introspection -- don't think about what YOU do" (Blumenthal). Watch real users try to figure out how it works. When one is running a study, just Smile and Nod! NEVER explain! (Enough said.) Write canned prompts: "What else did you expect to see? What do you want to do now?" Remember -- the software is being tested, not the user. The cardinal rule: "The user is always right!! There is no user error, just bad software."

3. The look and feel of the MARMAD/OptiX GUI constructed for beamline X12-C at the NSLS⁽¹⁾:

A fundamental structural feature of this software is modularity. These were all command-line-driven programs; we have produced a GUI that simply constructs text commands and pipes them to the program. In every case the original programs can still be run on their own -- with no GUI! This modularity leads to substantial flexibility. We have been able to patch in other detectors (Brandeis and ADSC CCD-based detectors) with very little modifications of either their or our software.

The data-collection hardware includes a MAR Research 300mm imaging-plate scanner mounted on the arm of an Enraf-Nonius FAST (CAD-4-style) diffractometer (Figure 1) and standard beam-line stuff -- alignment table, counters, and monochromator drive. The diffractometer is controlled by a many-process system:

- Diffractometer drive -- MADNES⁽²⁾. This is the same MADNES that was written by the EEC workshop, ran a diffractometer at LURE, the FAST, and the Argonne diffractometer at NSLS beamline X8-C.
- MAR detector drive -- MARdc. From MAR.
- Image-transformation process -- IPXform.

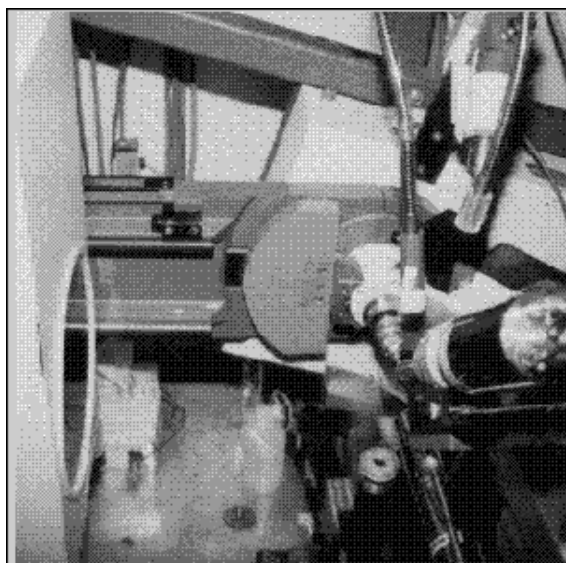


Figure 1: The hybrid MAR/FAST diffractometer at NSLS beamline X12-C.

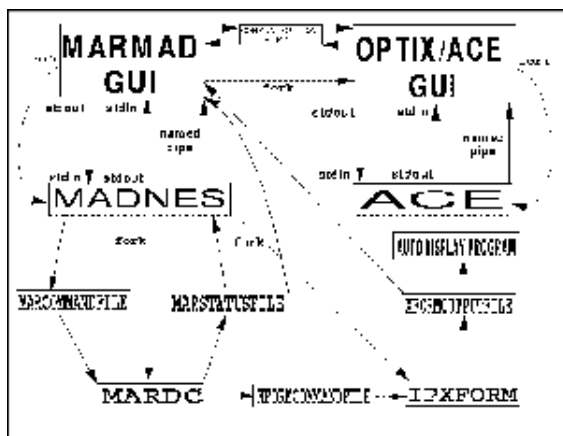


Figure 2: Logical connections among the beamline-control programs.

All of these are under control of a GUI.

Our GUI forks off MADNES, communicates with pipes that link standard I/O channels, and starts and communicates with other programs (Figure 2). The layout of the GUI matches the tree structure of MADNES, pruned to match the particular needs of our facility. Each level relates to different parts of data collection and analysis. There is a "standard" menu bar to give file control and to direct control to other levels. Other features include

- A scrolled text window for text output, including an adjustable paned window.
- Command line for conventional text input.
- Full frame adjustment to expand or decrease the size of the output window.
- Files menu that includes I/O of conventional MADNES files, Unix-style file commands, etc.
- Lots of color-coding of actions.

Levels that are most used are DET ("detector" or diffractometer immediate control) (Figure 3) and COLLECT. Both show Relative Zero parameters and status of the MAR scanner: a sliding bar to time exposures, erasures, etc., and color coding to show the step in the process. In the DET level one can set up the diffractometer, take test exposures, choose detector size: 180mm or 300mm, exert primitive control (scan, erase).

Control of the beamline comes from another combination of a GUI with an existing algorithmic program. The underlying program is ACE⁽³⁾, a program that allows control of stepping motors, counters, multi-channel analyzers, etc., mostly with command-line instructions.

With this program and the GUI we have constructed, called OptiX, we have reduced most beamline-control operations to a few button pushes. For example, ordinary beamline alignment requires only a button push to launch a few linked scans of the motorized table that carries the diffractometer, with optimisation logic providing the final position (Figure 4). A graph appears during the scan to show the profile of the x-ray beam.

Figure 1: The hybrid MAR/FAST diffractometer at NSLS beamline X12-C.

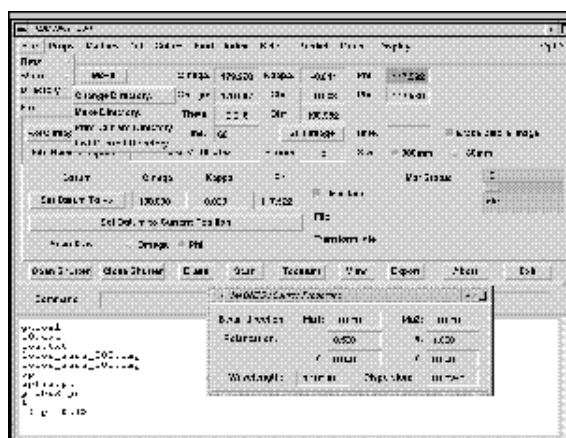


Figure 3: The immediate-control, or DET level of the MADNES GUI.

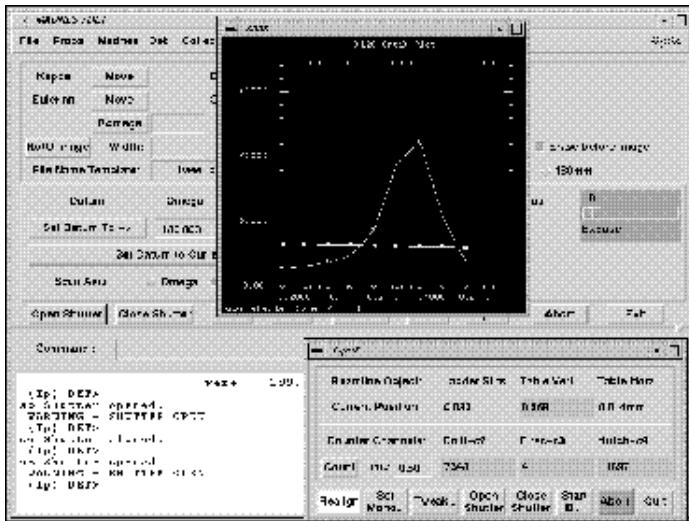


Figure 4: View of the OptiX GUI during realignment of the diffractometer to the x-ray beam.

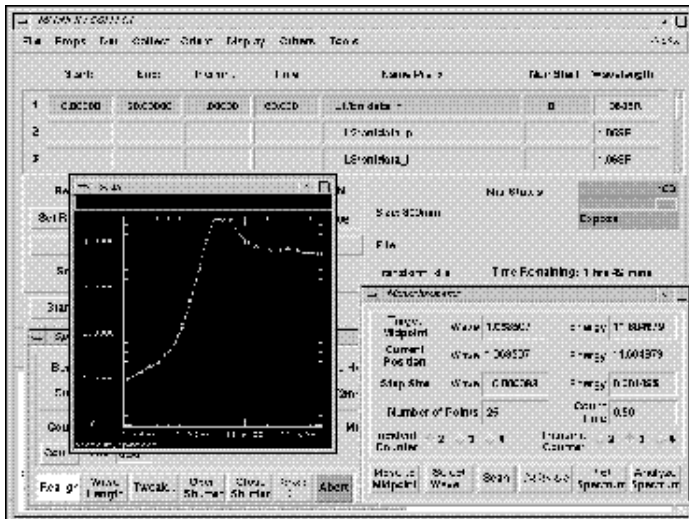


Figure 5: Automatic MAD data collection in progress with an absorption edge for a heavy atom displayed.

Sweeps of data are recorded with the COLLECT level of MADNES (Figure 5). There is a chart to set up multiple sweeps of data collection. This is useful for multi-wavelength data collection, to measure data for use of the anomalous effect by the "Friedel Flip" method, or to distribute data to several disks. For precise control of the monochromator for anomalous effects, we have linked control of the data sweeps with control of the monochromator. The monochromator setting in the COLLECT level is coded with a flag (R, P, F, A) to indicate whether the rising edge, the peak, the falling edge of the absorption spectrum of the heavy atom, or some absolute wavelength setting is to be used. The protocol of operation is that the monochromator scan is made before each data sweep. Then the spectrum is smoothed and the first derivative is calculated and either the maximum of the spectrum or a maximum or minimum of the first derivative is chosen and the monochromator is set to that wavelength. Finally the sweep of data is recorded.

4. Electronic logging of the experiment:

Three sorts of data are written "blindly" by the program to assure that the parameters of the experiment are safe. The output of the scrolled window goes to a file; a special "logfile" contains a concise textual report of operations made and data taken; a MADNES-style save file, containing all the settable parameters from MADNES is written with a coded name at the end of each sweep. Soon, each data file will contain a header with every possible parameter to describe the experiment stored in it.

5. Tools and toys:

The TOOLS menu contains a broadly useful tool to relate the resolution of diffraction seen on the detector with resolution of individual spots to the wavelength and specimen-to-detector distance. A widely used tool runs the **hkl** suite of programs (XdisplayF and denzo⁽⁴⁾) from within MARMAD to index an image and transfer the unit cell and crystal setting back into the control program. One then is able to run the ORIENT routine to select a particular orientation for the crystal for data collection. A Tactics tool allows one to choose the best rotation range(s) for a given crystal setting to get nearly complete data in the minimum of time. A logfile-editor allows emacs-style editing of the logfile so one can enter comments, etc. to provide an electronic textual record of the experiment.

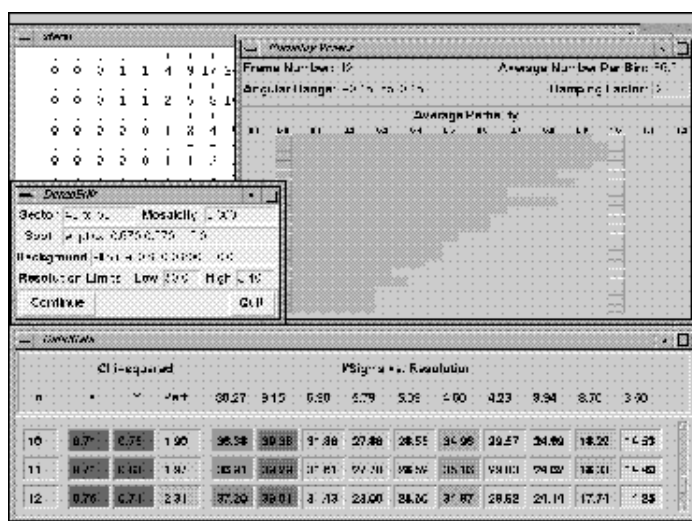


Figure 6: Output from onebut, the rudimentary automatic data-reduction link to the hkl suite (XdisplayF, denzo).

We believe that, to a first approximation, the experiment should be well enough defined by the known experimental conditions that data reduction should be automatic -- one should need to push only one button to accomplish it. Our first draft at doing this is shown in Figure 6. Here one can see two charts: a color-coded chart of I^2 values from the image refinement and of $I(I)$ values, and the histogram of partiality values from denzo, averaged over the last five images.

All our beamline documentation is available on the WWW: (<http://lsx12e.nsls.bnl.gov/x12c/x12cinfo.html>). It's especially useful for users to be able to study the beamline operating instructions before coming to BNL.

6. A summary of our experiences:

We have put a huge effort into the design, debugging, and continual maintenance of this software. There have been several goals: to assure the quality of the data, to relieve the training effort of the local staff, to extend the usefulness of the facility to relatively inexperienced users, to make life easier for the visitors to the beamline. It seems likely that the effort has been worthwhile, but there are still many things to do.

References:

1. J.M. Skinner, J.W. Pflugrath, and R.M. Sweet, *SHARE 80 Proceedings*, Session I224, (1993) San Francisco, Ca.

2. A. Messerschmidt and J.W. Pflugrath, *J. Appl. Cryst.* **20**, 306 (1987).
3. S. Kate Feng, D. Peter Siddons, Lonny Berman, "NSLS Beam Line Data Acquisition and Analysis Computer System". *Nuclear Instruments and Methods in Physics Research A347* 603-606 (1994).
4. Z. Otwinowski and W. Minor, "Processing of X-ray Diffraction Data Collected in Oscillation Mode" in *Methods in Enzymology* **276** (1997) C.W. Carter, Jr., and R.M. Sweet, eds. (in the press).