CONF-831203--143
UCRL- 89220
PREPRINT

UCRL--89220

DE84 012604

A New Kind of User Interface for Controlling MFTF Diagnostics

George G. Preckshot
Ralph A. Saroyan
John E. Mead

Proceedings of the 10th Symposium
on Fusion Energy
Fusion Energy Conference
Philadelphia, PA
December 5-9, 1983

November 29, 1983

Lawrence Livermore National Laboratory

DISCLAIMER

A NEW KIND OF USER INTERFACE FOR CONTROLLING MFTF DIAGNOSTICS

G. G. Preckshot, R. A. Saroyan, J. E. Mead
Lawrence Livermore National Laboratory
P. O. Box 5511 L-535
Livermore, CA 94550

Abstract

The Mirror Fusion Test Facility (MFTF) at Lawrence Livermore National Laboratory is faced with the problem of controlling a multitude of plasma diagnostics instruments from a central, multiprocessor computer facility. A 16-bit microprocessor-based workstation allows each physicist entree into the central multiprocessor, which consists of nine Perkin-Elmer 32-bit minicomputers. The workstation provides the user interface to the larger system, with display graphics, windowing, and a physics notebook. Controlling a diagnostic is now equivalent to making entries into a traditional physics notebook.

"Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48."

Introduction

In 1986, MFTF-B is scheduled to go into operation as a fully computer controlled experiment operating on a five minute cycle. The machine will be operated by a crew of machine operators and physicists, with the physicists being responsible for comprehending immediate results and directing machine control to achieve an experimental plan. To assist physics personnel in reaching their goals, a user interface has been designed to control MFTF-B diagnostics and to provide timely presentation of crucial experimental data.

In this paper we will discuss the environment which constrains the user interface, the tasks the user interface must support, and some details of the user interface itself.

The Environment

The computer environment of MFTF-B consists of a multiprocessor (nine 32-bit minicomputers) connected to the experiment through many LSI-11 microcomputers and to users through a variety of consoles, terminals, and intelligent workstations. Detailed descriptions may be found elsewhere; a companion paper [1] presents the computer hardware structure and data rate calculations for data movement within the MFTF-B plasma diagnostics system. It also includes a list of references which describe other components of the MFTF-B Supervisory Control and Diagnostics System (SCDS).

There are additional non-computer environmental factors of importance to the user interface design. In particular, the machine operating scenario, the large amount of data acquired, and the short time between experimental cycles strongly affect the ways users must interact with the system.

In the operating scenario proposed for MFTF-B, a "shot"(1) may occur once every five minutes under the cooperative control of a physics shot leader (in charge of the physics crew) and a chief operator (in charge of the machine and a crew of operators). Up to eight megabytes of data will be acquired. A shot may last as long as 30 seconds and data transfer may take as much as one minute after a shot. In addition, setup time may be required for the next shot. Therefore only a short time between shots is available for human comprehension, decision making, and decision implementation.

The environment we have described severely constrains the plasma diagnostics system in general and the user interface to the diagnostics system in particular.

Five Major Tasks

We have identified, in order of priority, five major real-time tasks required of the plasma diagnostics system. They are:

1. Acquire and save raw data within one shot cycle.

2. Rapidly present a subset of acquired data to allow experimenters to make operating decisions.

3. Permit interactive adjustment of diagnostic equipment and real-time changes of data acquisition parameters.

4. Perform modest numerical processing and present results in time to permit experimenters to make medium term machine operating decisions (over a course of several shots). Provide means for experimenters to change processing specifications on a short or medium term interval.

5. Provide means to retrieve saved experimental data for use locally (by a background task) or on other computer systems.

The user interface figures prominently in every item but the first. However, even the first item has a profound effect upon the user interface, since an incorrect order of data acquisition could cause detrimental delays in presentation of data required for decision making.

The Diagnostic Desktop Metaphor

Several successful user interfaces have a central theme or metaphor which helps the user form a mental model of the system. The Xerox 8010 Star Information System [2], for instance, uses an office metaphor complete with desktop, file folders, and filing cabinets. The Apple Lisa "Desktop Manager" [3] implements a similar metaphor.

The MFTF-B plasma diagnostics user interface is also based upon a desktop metaphor. Unlike the business office desktop, however, the diagnostics desktop may contain such graphics "objects"(2) as diagnostic control tables, data processing commands, graphical representations of results, and an experimental notebook. See Figure 1 for an example of such a desktop. Some objects (such as the notebook) may be composed of others.

---

(1) A "shot" consists of particle injection, density buildup and a plasma heating cycle or cycles. Shots can last up to 30 seconds in MFTF-B.

(2) An "object" is a table, a graph, a page, or any sort of association of text and graphics for which there are well defined rules for manipulation. For instance, from a graph you may pick off X-Y values. In a table, X-Y value pickoffs don't make sense but changing particular table entries may.

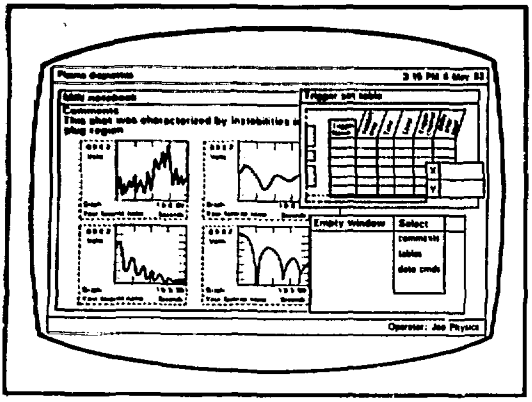

Figure 1. Typical diagnostics desktop

In both the Xerox Star and the Apple Lisa user interfaces, the user employs some sort of pointing device to manipulate graphics objects in "windows"[3]. A very popular pointing device, which we favor, is called a "mouse". Our personal experiences with this device, as well as reports of others [4], indicate that users very rapidly develop good hand-eye coordination and can make selections quickly and accurately. This speed is crucial for the MFTF-B diagnostics system because of the limited time available for implementing decisions.

A user manages graphics objects with menus, which appear on demand. Menus allow selection of various alternatives and general manipulations of the workspace (such as defining or moving windows). Objects may be selected, moved into windows, and manipulated. In what follows, we will describe some of our major objects and how they are used to accomplish the goals of MFTF-B diagnostics.

Such desktop emulators are a tremendous interrupt and graphics processing load for a computer, and thus are not compatible with simultaneous real-time data acquisition. Hence, we have placed the interactive portion of the user interface in commercially available graphics workstations attached to our computer system by parallel links.

Data Acquisition

Because of the importance of acquiring the raw data from the experiment, we have separated the actions required to specify details of data acquisition from those actions required to display or process the acquired data. Our intention is to avoid distracting the user with details about what to do with the data while the user is deciding how to acquire the data.

The physicist specifies which data he wishes to acquire by making entries into tables. An example of such a table is shown in Figure 2a, a probe table which reflects the electrical schematic shown in Figure 2b. For example, the user may wish to change the RF probe assembly attenuation (indicated by the arrow in the figure). The user moves the arrow into the table entry by moving the pointing device. Once the user has selected an entry by pointing at it, that entry may be changed using the keyboard (or other method). This change will be sent to diagnostic hardware either immediately or at a later time, depending upon user

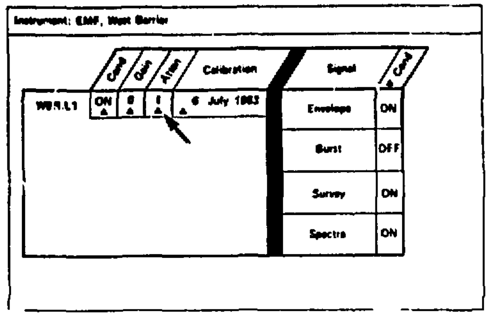

Figure 2a. RF Probe Control Table

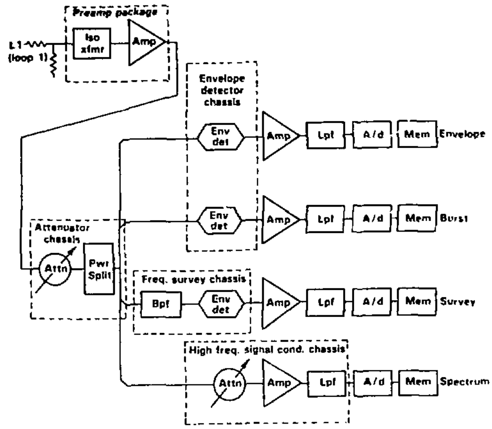

Figure 2b. RF Probe Schematic (Courtesy Dave Goerz)

selection. Table entries may, therefore, act like knobs connected directly to diagnostic hardware.

The diagnostic system maintains some table entries. For instance, the system can sense which data recorders are plugged into a probe assembly and thus add the "signal" column when a recorder is connected. The tables therefore serve for both control specification and status presentation. Since tables are automatically recorded in the diagnostic notebook at the completion of each shot, they also serve as a record of data acquisition settings.

We chose a table presentation for data acquisition control over other graphical representations for several reasons. The success of electronic spread sheets such as VisiCalc[4] demonstrates that tabular presentations are easy and convenient to use. Graphical representations of knobs and buttons (such as those used in the MFTF-B facility control consoles) are not convenient forms of archival record for data acquisition setups. Tables, on the other hand, are convenient forms. Since we feel that the user should not have to shift mental gears when looking at control information on a workstation screen and previous control information in a notebook, we chose the tabular representation.

---

(3) A "window" is a rectangular subset of the area on a CRT screen. Such windows may individually act like smaller CRT screens, giving the appearance of multiple concurrent displays.

(4) VisiCalc is a trademark of the VisiCorp Corporation.

Our tables have additional capabilities. Some table entries may lead to additional information. For instance, to find the CAMAC data recorder used to record the "envelope" signal in Figure 2a, the pointer is moved into the "envelope" table entry. "Picking" it (by pressing a button) causes the associated recorder table to appear. The set of control tables reflects the physical design of the diagnostic equipment. Users may elect to see as much or as little information as they wish about their diagnostic data acquisition equipment.

When the experimentalist has filled in the data acquisition tables (either by interactive entry or by directing the system to use tables from previous experiments), he submits them for use during an experiment.

Data Processing and Display

Because of the division between data acquisition specification and the specification of which data to process and display, the experimentalist may delay decisions about which data to display until well after automatic shot cycling and data collection have commenced. We expect that normally only a subset of data will be processed or displayed during active machine shot cycles because of the large amount of data being collected. As previously mentioned, there are two objectives to be satisfied: rapid presentation of some data for immediate experiment guidance and medium term presentation of modestly processed data for longer term experiment guidance. We use "data commands" to specify processing, display, or movement of data.

Data commands, or sets of them, actually comprise a simple language which can be described by a BNF-style grammar. The language has some similarity to the Unix "pipe" concept [5] and is used to specify a network of *multiple-input/multiple-output processes* (the nodes) connected by sets of data (which form the arcs of the network).

Data commands are displayed as sentences which describe data sets and the processing or movement to be performed on the data sets. To create a data command, the user may type it, as is common in other computer languages, or the user may pick the elements of the data command from menus. Menu selection has advantages, since only syntactically correct choices are displayed. Menus also offer the opportunity to present additional information or help relating to particular menu choices. Example data commands are shown in Figure 3.

One of the methods of displaying data processing results (or, indeed, raw data) is to specify a data destination called a "graph". Figure 3 has several such data destinations. A graph, once produced, may be manipulated interactively. A user may pick values from a displayed graph with a cursor, enlarge part of a graph, hardcopy the graph, or copy graphs to or from an experimental notebook.

To achieve our objective of rapid presentation of some data (termed "quick-look" data) for immediate experiment guidance, data commands have a data destination called "quick". Figure 3 has an example. Specifying a "quick" destination has the same effect as specifying a graph except that it changes the order of raw data arrival. Quick graphs are produced first since the raw data required to produce them arrives first. We expect quick-look data displays to be available on the order of ten seconds after shot completion, giving experimentalists as much decision time as possible.

Retrieving and Using Saved Data

Most of our users will be physicists, and most physicists are acquainted with laboratory notebooks. To provide a familiar interface to data stored in the data base [6], we have chosen

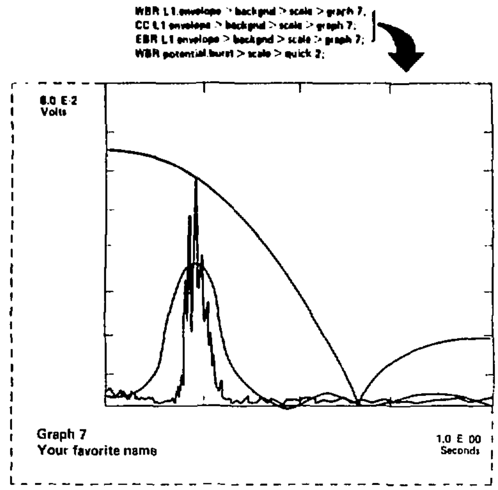

Figure 3. Data commands, showing three signals going onto graph 7 and one quick-look signal.

to present the data base in the guise of an electronic analog of a paper laboratory notebook.

The electronic notebook paradigm has important differences from its paper antecedent. The most notable difference is that diagnostic switch settings, gains, and data are automatically recorded in the electronic notebook during shot cycles. Because it is electronic, old setups and data commands may be read directly by the system and used in new experimental setups.

There are many similarities, however. The user may manually enter (or cause to be entered) a variety of items, such as comments, new data commands, and additional processing results. The notebook itself appears to be composed of ordinary pages which may be viewed on a high-resolution graphics screen or hardcopied. When viewed on a graphics screen, the notebook can be browsed, page by page. In addition, there are tables of contents and indices which allow the researcher to find things just as he would in a paper book, but unlike a real book, the user may make new entries in the indices. The table of contents helps the user to find tables, graphs, data commands and other pre-defined objects. The indices, on the other hand, permit the user to mark pages with user-defined keywords or key phrases. An index is used much the same as an ordinary book index, with alphabetic keyword browsing being one method.

Both the table of contents and the index permit the user to locate data or references to data that are of particular interest. Data are either available in immediate form (as a graph or table) or in indirect form as a name. Data acquisition tables, which are automatically appended to the notebook whenever data is acquired, include both a status (the result) and the name by which the data is known. The naming system reflects the physical construction of the experiment. To use the data in any numerical process, the researcher refers to it by name. Thus the notebook is the experimenter's interface to data stored in the database.

Conclusion

The user interface to the MFTF-B plasma diagnostics system is based upon an experimental desktop metaphor. It is presented to the user by microprocessor-based interactive graphics workstations, which act as peripherals to our main system of interconnected 32-bit minicomputers. The experimentalist controls his diagnostic and data processing by manipulating control tables and data commands, which automatically become part of an electronic notebook. The researcher uses the notebook as the gateway to data and results stored in the data base.

Acknowledgement

We appreciate the helpful comments and suggestions of our colleague V. L. Renbarger during the preparation of this paper.

References

1. Wyman, R. H., et al, "An Overview of the Data Acquisition and Control System for Plasma Diagnostics on MFTF-B", PEPFR 10, December, 1983.

2. Smith, D. C., Harslem, E., Irby, C., Kimball, R., "The Star User Interface: An Overview", Proc. of National Computer Conference, June 7-10, 1982, Houston, pp. 515-528.

3. Promotional Literature, The Apple Computer Corporation, 1983.

4. Smith, D. C., Irby, C., Kimball, R., Verplank, B., Harslem, E., "Designing the Star User Interface", Byte, 7 (4) April 1982, pp. 242-282.

5. Ritchie, D. M., Thompson, K., "The Unix Time Sharing System", Comm. ACM 17, July 7, 1974, pp. 365-375.

6. Choy, J. H., Wade, J. A., "A Data Base Management System for the MFTF", PEPFR8, November, 1979, pp. 942-945.