TELEMETRY SYSTEMS FOR THE 90's:

GRAPHICAL USER INTERFACES WITH PROGRAMMABLE BEHAVIOR

JONI VELASCO

Senior Member of the Technical Staff
Computer Sciences Corporation
Realtime Data Systems Center
301 North R Street
Lompoc, CA 93455

ABSTRACT

The design and development of user interfaces for telemetry data processing systems is undergoing a period of rapid change. The migration to graphics workstations is raising expectations and redefining requirements for user interfaces in the nineties. User interfaces which present data in crude tabular form on alphanumeric terminals are on a path to extinction. Modem telemetry user interfaces are hosted on graphics workstations rich with power and software tools.

This paper summarizes the evolution of user interfaces for telemetry systems developed by Computer Sciences Corporation, highlighting key enhancements and use of third-party software. The benefits of prototyping and the trend toward programmable interface behavior are explored.

KEY WORDS: Graphical User Interface
Workstation
Prototyping
Programmable User Interface

INTRODUCTION

The Realtime Data Systems Center (RDSC) of Computer Sciences Corporation (CSC) has designed and implemented user interfaces for telemetry processing systems for over a decade. The design of those interfaces has been influenced by end user requirements and current technologies. With each new system, CSC strives to develop a telemetry user interface which provides powerful graphics and maintains a careful balance between simplicity and flexibility.

The fundamental role of the user interface in a telemetry system is to allow users to monitor and analyze data during flight test and missile launch operations. Innovations in hardware and software technologies have greatly improved the effectiveness of data presentation over the past decade. In early systems, data was presented in tabular form on alphanumeric terminals. Analyzing data was difficult on screens full of digits. Modem telemetry systems present data to users with full color graphics making key events and anomalies easy to recognize.

As telemetry users become increasingly computer literate, their user interface requirements have become more stringent. Users expect the interface to offer a “look and feel” equivalent to that enjoyed in the personal computing environment. A windowed environment with a Graphical User Interface (GUI) is a new requirement being levied on user interfaces today. In general, the trends laid down in the commercial computing industry often set the stage for future telemetry system user interfaces.

This paper portrays the evolution of user interfaces for telemetry systems built by CSC over the past decade. The features of each user interface are illustrated through its benefits to the end user.

Telemetry User Interfaces: Then and Now

The evolution of telemetry system user interfaces built by CSC has been guided not only by technological advances in the computer industry, but also by careful cost considerations. The competitive nature of fixed-priced procurements compels CSC to create user interface solutions that are a compromise between capability and cost. Because the cost of workstations was prohibitive until the late 1980’s, most user interfaces were implemented on graphics terminals directly connected to a host mainframe in a non-distributed architecture. During the late 1980’s, significant performance boosts and dramatic price reductions made workstations a viable component of distributed systems.

The software available with most graphics terminals used in early systems was scarce and of a low level nature, requiring CSC to allocate considerable time and money to new graphics application development. Conversely, workstations appeal to such a broad market that a vast amount of graphics software with enormous off-the-shelf (OTS) capability is readily available. The cost of developing a graphics capability equivalent to that offered by today’s products is typically prohibitive. With the time constraints imposed by an aggressive development schedule, the integration of OTS graphics products today is a very attractive option for integrators such as CSC.

In addition to reducing the software development effort, the integration of OTS graphics software minimizes the risks inherent in new code development. Selection of a reputable vendor with an outstanding reputation for customer support is essential when choosing an OTS product. As an added benefit, most vendors provide periodic software updates which correct problems and often broaden the functionality of their products.

Resume of Telemetry User Interfaces

Over the past ten years, CSC has built systems which offer the end user an increasingly more sophisticated user interface (Figure 1). From early systems which offered little or no visual prototyping, to contemporary systems which give users the utmost creative control, user interfaces have matured rapidly over the last decade.

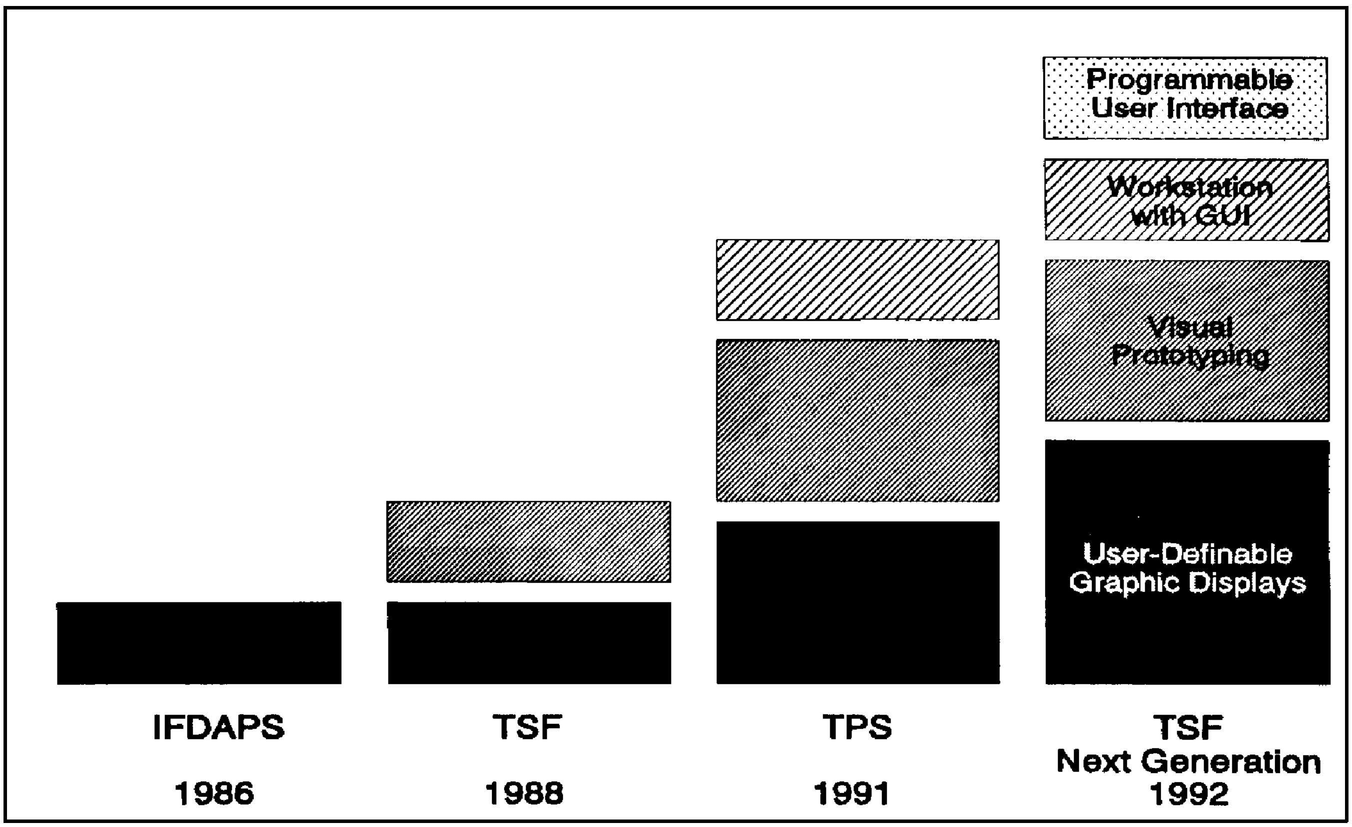

Figure 1. Evolution of Telemetry User Interfaces. A representative collection of projects highlight the improvements of each user interface implementation (or development).

Integrated Flight Data Processing System (IFDAPS)

The IFDAPS was developed from 1981 to 1986, before low cost graphics workstations burst onto the scene. Using a traditional non-distributed architecture, the IFDAPS user interface for realtime data monitoring is provided on Megatek 7500 graphics terminals directly connected to a host computer.

The approach for defining display formats in IFDAPS is analogous to the manner in which programmers develop code. IFDAPS users design displays on paper and then describe them through a high-level language called IFDAPS Source Language (ISL). The ISL is processed by a special compiler and stored into object files. The IFDAPS must be initiated before the displays can be viewed, often resulting in contention for resources between setup and realtime system users.

Through ISL, IFDAPS users define displays comprised of stripchart graphs, bar charts, tabular data, and special artificial horizon graphs (Figure 2).

The IFDAPS displays data with graphics, but user commands are entered as text. A comprehensive GUI, with user input through graphical objects, was neither an expectation nor a requirement when IFDAPS was developed.

Though IFDAPS does offer the user some flexibility in creating custom screen formats, the absence of visual prototyping is time consuming and hinders the display development process.

Test Support Facility (TSF)

The TSF system, developed from 1985 to 1988, was based on IFDAPS with a similar hardware platform hosting the telemetry user interface.

The TSF greatly improved the IFDAPS user interface through development of a graphical editor referred to as the Interactive Display Generator (IDG). IDG was developed to provide rapid prototyping of custom displays with immediate visual feedback. Display entities are sized and placed with a point-and-click interface via a trackball device. The IDG reads and writes display descriptions to object files which are used directly by the TSF system.

The TSF display capabilities transcend IFDAPS through realtime display modifications made possible with IDG. At the request of a user, a display can be rapidly edited and made available for use.

This streamlined approach to display development encourages an ongoing cycle of enhancements and responsiveness to user needs at TSF. While the GUI and prototyping offered by IDG did break new technological ground, its fundamental display capabilities were based on IFDAPS with its inherent limitations.

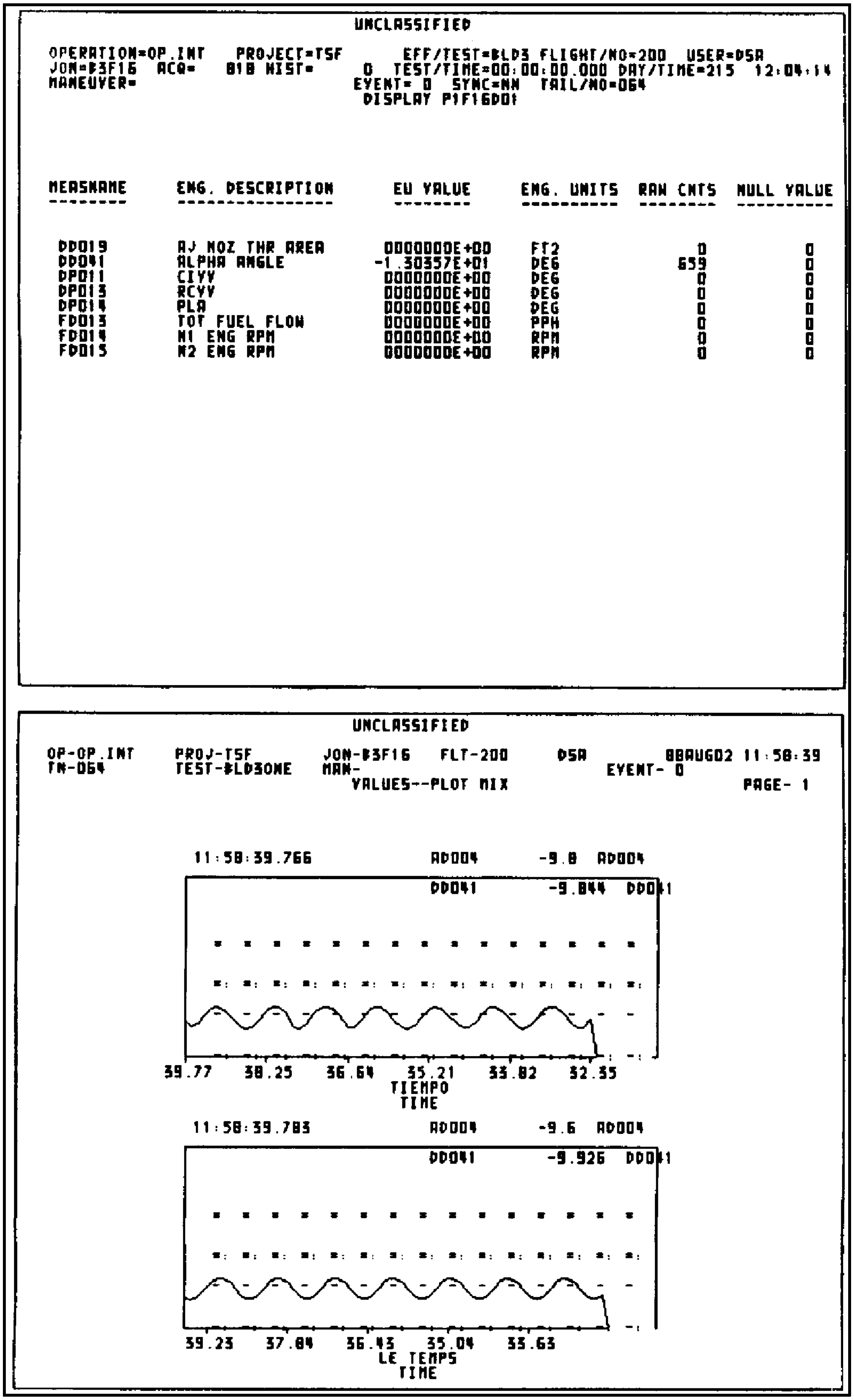

Figure 2. Sample IFDAPS Displays. The object types and available screen layouts are limited in IFDAPS.

Telemetry Processing System (TPS)

A distributed system architecture characterized TPS, developed from 1988 to 1991. At the time this system was designed, workstations with marked performance gains and attractive pricing had become a reality. Enhanced product lines were introduced at an extremely brisk pace, causing prices of existing equipment to spiral downward. CSC took advantage of this opportunity and designed the TPS to include data distribution to an array of VAX 3200 workstations via Ethernet.

The graphics capabilities of the workstation warranted a fresh look at the design of the telemetry user interface. For the first time, a number of third-party software products were available which would provide significant off-the-shelf capability and reduce project development costs. After careful industry review, the DataViews product from V.I. Corporation was selected. DataViews includes a suite of tools allowing CSC to provide a sophisticated graphics user interface with minimal code development.

The graphics software organization reflects the tremendous OTS capabilities offered by the DataViews product (Figure 3).

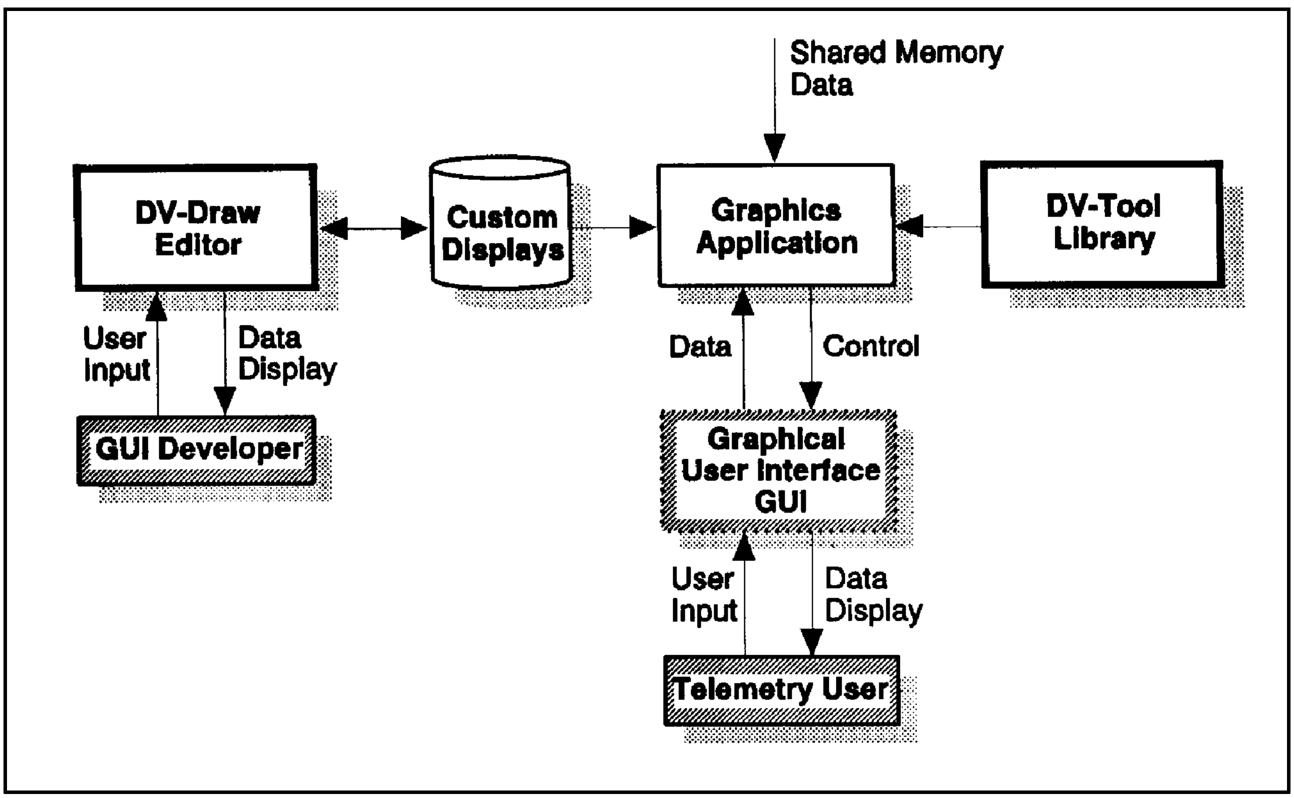

Figure 3. Graphics Software Organization. The DataViews product offers a wealth of off-the-shelf capability.

The DV-Draw graphical editor allows users to prototype displays through an intuitive point-and-click interface. The user constructs a custom display by choosing and placing objects in the drawing area. Over a hundred display objects are available including simple objects such as circles and rectangles, graphs and charts, and text (Figure 4). Displays can be previewed in DV-Draw with dynamic updates using simulated data. Custom displays are saved on disk for later use by the graphics application.

The application calls DV-Tools library routines to perform a host of graphics tasks. DV-Tools routines give the application control of the user interface, without requiring low-level graphics functions to be implemented. The application calls DV-Tools to load displays, connect display objects to actual data, and perform cyclic update of the screen.

DV-Tools and input objects enable the application to present a GUI, complete with graphical input, with a minimum of effort. User input is collected through graphical objects such as menus, toggles, checklists, and sliders.

Test Support Facility - Next Generation

The next generation TSF system employs a distributed architecture with a host computer distributing data to an array of Silicon Graphics Incorporated (SGI) Personal Iris workstations via the shared memory Universal Memory Network (UMN) developed by CSC.

Resembling the TPS implementation, the DataViews product provides the foundation for graphics on the TSF workstation. The next generation TSF extends the graphics capabilities by giving the user the ability to define the telemetry user interface. In older systems, users built custom displays but development of the user interface was a significant programming task for CSC. Graphical tools allowed users to define the appearance of a user interface, but lacked a means by which the behavior could be described, forcing the behavior of the interface to be hard-coded in the application. The implementation of rules in DataViews 8.0 paved the way for an innovative application design.

The DV-Draw user defines interface behavior in terms of rules which are comprised of events, conditions, and actions; and then runs a prototyping option to simulate the behavior of the interface. The Realtime Executive (RTX) application was developed by CSC to exploit the programmability of rules. In the same way the rules are processed by the DataViews prototyper, RTX reads and processes rules to control interface behavior at runtime. For example, a rule might specify that when an object is “picked” (event) with the right mouse button (condition), a new is view loaded (action). This example illustrates how the user can control the transition from one screen to another.

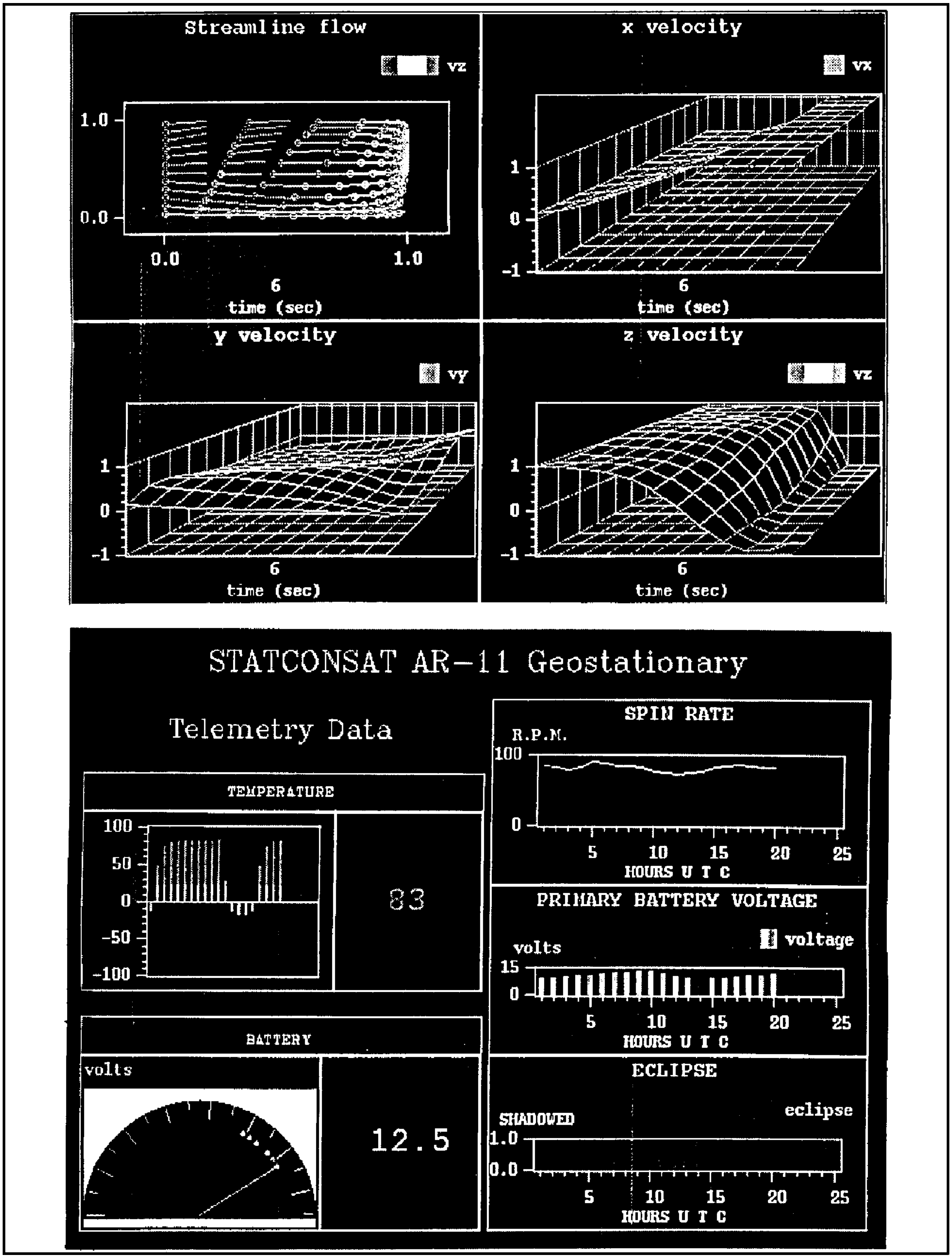

Figure 4. Sample DataViews Displays. The format of DataViews displays are limited only by the user's creativity.

The functionality of DataViews has been extended by RTX through the definition of keywords which are entered by the user in DV-Draw. These RTX keywords are entered in text fields and support enhanced data definition, processing, and many other special features. RTX keywords give the user control over display and object update rates, and allow the user to define drawports (i.e. drawing areas) and define a projection type for the drawport. Drawport projection determines whether or not the aspect ratio of a display will be maintained when it is drawn.

The Data Definition Language (DDL) is used to request access to data in memory. Each data item accessible through DDL is referred to as a DDL source and is solicited with a unique DDL keyword entered in DV-Draw. The keywords which comprise the DDL are uniquely defined for each project. In TSF, the DDL provides access to measurement data, measurement descriptions, date and time information, telemetry hardware status, and many other types of shared memory data.

The RTX and its innate support of a programmable user interface has revolutionized the way CSC's customers participate in the development of the telemetry user interface. The user has complete creative control of the user interface from concept definition through implementation and maintenance. RTX is a generic application which can support an infinite number of user interfaces, each with a unique appearance and behavior. The benefit of this implementation will be realized with future projects developed by CSC. RTX is non-project specific and provides a graphics capability which will be exploited by future systems.

CONCLUSION

Telemetry user interfaces are maturing at an energetic pace. The appearance of workstations has given these interfaces a whole new look. Once rare, a telemetry user interface with a fully windowed environment and a GUI is becoming standard. Users will accept no less. No longer satisfied with a programmer's approach to defining displays through descriptions, users want a graphical tool which provides visual prototyping for rapid development of displays.

The most recent and exciting characteristic of telemetry user interfaces is the trend toward programmability. A programmable user interface shifts control of the interface behavior from the application to the user. Aside from the obvious benefit to the user, applications which extend programmability to the user are generic and thus portable. Programmable user interfaces effectively support the dynamic needs of users and will no doubt become a requirement for future telemetry systems.

ACKNOWLEDGMENTS

A special recognition belongs to Philip Price, whose creativity and vision have contributed greatly to inventive concepts such as the RTX.

REFERENCES

Computer Sciences Corporation: The Realtime Executive Users Guide

V.I. Corporation: DataViews Documentation Set, Release Number 9.0, June 1, 1991

NOMENCLATURE

CSC	Computer Sciences Corporation
DDL	Data Definition Language
GUI	Graphical User Interface
IDG	Interactive Display Generator
IFDAPS	Integrated Flight Data Processing System
ISL	Interactive Source Language
OTS	Off-the-Shelf
RDSC	Realtime Data Systems Center
RTX	Realtime Executive
SGI	Silicon Graphics Incorporated
TSF	Test Support Facility
TPS	Telemetry Processing System
UMN	Universal Memory Network