"MY USER INTERFACE IS THE BEST BECAUSE..." (PANEL)

Moderator: A. Brady Farrand, Relational Technology
P.O. Box 4006
Alameda, CA 94501
415-748-2751, brady@hare.rtech.com

Panelists: Tom Erickson, Apple Computer
Tony Hoeber, Sun Microsystems
Bill Parkhurst, NeXT Inc.
Ted Wilson, Hewlett-Packard and the Open Software Foundation

INTRODUCTION

Brady Farrand
User interface architect for Relational Technology, makers of the INGRES™ database. His company is faced with the choice of implementing their product under one or more of these user interfaces. He was once the guardian of the user interface standards for Xerox's Star/Viewpoint™ software.

In the marketplace and in the media, the major players in the computer industry are battling to make their choice of a direct manipulation user interface for their systems into the "standard" user interface for all systems.

Since CHI '88 met in May of 1988, less than a year ago, several user interface products have appeared with the strength to fight for preeminence. AT&T and Sun Microsystems put together a joint software development effort that created Open Look®, a direct manipulation face for UNIX®. In response, many of the other major Unix system houses banded together to form the Open Software Foundation. OSF composed an alternate "standard" direct manipulation face for Unix, OSF/Motif™. NeXT® not only introduced with much media hoopla a truly object-oriented user interface, they licensed it to IBM where it will become the face for some of the computer giant's Unix offerings.

The panelists represent four major competitors for the emerging "microstation" market (systems powerful enough for business in the 90's yet small enough for individuals to use). The microstation market is driving the innovations in computer software. Unix, OS/2® (and its predecessor, MS-DOS™), and Macintosh-OS™ are the dominant operating systems for this market. Innovations in user interface design for these operating systems and the applications that run on them are leading the way into the 90's.

The panelists do not represent all the competitors. They do represent four products with distinctly different sets of appearance and functionality, the defining features of a user interface. Each product is contributing innovations to user interface design. By better understanding these four products we can better understand four different approaches to a direct manipulation interface. Years from now, we will be able to look back and judge which approach was validated in the marketplace. Macintosh has already made the direct manipulation interface popular and familiar to the public. Open Look and OSF/Motif are independently designed, with different looks and different ways of doing things, to perform similar tasks for a Unix machine. They will soon be evaluated by the marketplace. NeXT has created a face for Unix that takes advantage of an object-oriented programming environment and a high resolution display to give its user interface a distinctly different flavor.

Left out are several major user interfaces less distinctly different from the rest. Most notable are Microsoft's Presentation Manager™, Hewlett-Packard's New Wave™, and Digital Equipment Corporation's DEC-Windows™. They all contributed to OSF/Motif, thus blurring the distinctions between the products.

The panelists are proud parents with beautiful babies. They want the rest of the world to appreciate their offspring. They will briefly describe the look and feel of their products. But most of all they will describe what is special about their products. They will focus on the things that make them proud of the user interface. We will get a chance to look at the user interface through their eyes, and maybe better understand its character and its design.

For the developer two sides of the user interface product are important in selecting it to be the face for an application. The first is its "ease of use," how the look and feel simplifies the end user's job, working with the application. The second is "ease of implementation," how the product's

tools simplify the programmer's job, adding that user interface to the application. Both are important. Good user interface never becomes part of the product unless it is easy for the programmer to make it part of the product. Good tools even promote good design. One key difference between the panelists may be the emphasis on "ease of use" or "ease of implementation" issues.

The panelists are here to champion their user interfaces. Will one of these user interfaces emerge and dominate the others? Will they try to be truly universal by being implemented on top of other operating systems? Why would one user interface win out over the others?

The panelists will, in turn, state why they believe their user interface is the best of the bunch. They will then be given an opportunity to take issue with the others' claims. Finally, the audience will be given an opportunity to question the panelists.

PANELISTS' STATEMENTS

Tom Erickson
Human interface engineer at Apple Computer, in charge of issuing the Apple Human Interface Guideline updates and responsible for refereeing internal debate regarding those guidelines. For the last six years Tom has designed and written user interfaces (and their specifications) for commercial products on DOS, Microsoft Windows, and Macintosh systems. Prior to that he did graduate work in Cognitive Psychology at the University of California at San Diego, where he was a member of Don Norman's Human Machine Interaction group and the Cognitive Science Laboratory.

There are three reasons why Macintosh has the best human interface. First, the Macintosh human interface guidelines are user-centered. The guidelines not only describe what to do, but, more importantly, they describe the consistently user-centered rationale behind the guidelines. Because no user interface specification can specify everything, it is essential to make the reasons behind the guidelines inescapably clear.

Second, the design of the Macintosh human interface does not stop at the edges of the screen-- it extends to often neglected components of the human interface: documentation and hardware. The ease-of-use of Macintosh documentation and hardware are important parts of the user's experience.

Finally, the Macintosh interface is the best because there is an interface-sensitized culture that supports it. End users and reviewers scream if products do not follow the Mac interface. Macintosh developers are strongly committed to providing easy-to-use interfaces. And Apple as a corporation is committed to having the best human interface possible.

Tony Hoeber
Tony is the leader of the Open Look design team. He has worked at Sun for almost 5 years, concentrating mainly on the SunView user interface toolkit and on Open Look. He is the author of the SunView™ Programmer's Guide and the NeWS™ Technical Overview.

Open Look represents the most mature and well conceived user interface on the market. It contains many innovative features, such as the pushpin for pinning up menus and windows for repeated use, a coherent way to use color that is both attractive and ergonomically sound, and a simple, one-step way to scale any window to accommodate different screen resolutions, eyesight, viewing conditions or tasks.

But Open Look's most significant innovation lies not in any particular feature, but in the fact that it is the first user interface to be designed from the start to be independent of any particular implementation. Open Look is specified in device independent terms, so that it takes best advantage of the full range of displays, keyboards, and mice. This means that Open Look is not tied to any particular piece of hardware, operating system, window system, or toolkit.

The design of Open Look represents not a patchwork of features from other systems, but a coherent design based on clear principles. The design goals were: 1) a clean, understated, attractive visual style, 2) the proper balance between the attributes of simplicity, consistency and efficiency, 3) device independence, and 4) interoperability with the other popular user interfaces.

Bill Parkhurst
Software Manager at NeXT, Inc. At NeXT for three years, he contributed initially to the system architecture and later led the group responsible for specifying and implementing the user interface via the Application Kit™ and Interface Builder™. Prior to NeXT, Bill spent two years as an independent Macintosh developer.

The user interface on the NeXT Computer is the best because from the very start of the computer's design, the

user's needs were kept in mind. NeXT has a strong corporate commitment to excellence in its user interface. We have designed our user interface to match a sophisticated hardware and operating system environment consisting of a large screen, a large file system and concurrent processes. In addition to blending and refining best ideas in user interface design, we have added new, innovative elements such as our Application Dock and Directory Browser. Consistency of our user interface is virtually guaranteed across applications. We provide application developers with powerful tools, the Application Kit and Interface Builder, which allow a conforming user interface to build with virtually no programming. Finally, our user interface is very aesthetically pleasing. By appropriate use of highlights and shadows, we have created a look with depth and realism.

Ted Wilson
Ted led the team that selected the User Environment

Component for the Open Software Foundation. Currently, he is manager of research and development for Hewlett-Packard's Corvallis Workstation User Interface Laboratory.

On December 30, 1988, the Open Software Foundation announced the selection of the graphical user interface technologies that will be used to form its core User Environment Component (called Motif). Selection of these technologies was based on technical excellence, maturity, support of standards, ability to perform in a heterogeneous network environment, and compatibility with PC's.

Motif is an optimum composite of the technology acquired by OSF from the Digital Equipment Corporation and the Hewlett-Packard company, and includes a toolkit of user interface components, a presentation description language for specifying interface components, a window manager to enforce user interface policy, and a style guide.