



From the Editor in Chief...

## Interface Pains

Robert E. Filman • RIACS/NASA Ames Research Center • [filman@computer.org](mailto:filman@computer.org)

Last week I tried to reserve an auto rental over the Web. The site asked me where and when I wanted to rent, presented choices of possible cars, went through a long dialogue to get personal data such as addresses and credit-card numbers, and then told me that the car I had selected was unavailable. I iterated the process through several different types of cars, always with the same result. I eventually retreated to an albeit-more-expensive-but-nevertheless-willing-to-actually-rent-me-something competitor. I'm not sure what the original rental company thought this activity's purpose was, unless it hoped to catch me failing to uncheck the "send me email offers" box, thereby providing a "requesting" spam recipient into perpetuity.

At last month's European Conference on Object-Oriented Programming (ECOOP), I met a student contemplating a dissertation on graphical tools for program understanding. She expressed the belief that the best tools maximized the amount of information presented to users. I was surprised, after 20 years of Tufte<sup>1-3</sup> and work on the theory and practice of interface design, that a student building a system with a major interface component would start from such an assumption.

These experiences lead me to believe that a discussion on the design of usable systems might be appropriate.

I told the student that maximizing the amount of information a system conveys to a user isn't a good idea. Rather, the designer should keep in mind what the user is trying to accomplish and figure out how to minimize the effort (physical and intellectual) the user must expend to get that task done. For example, if a user is seeking a single value, it's better to present a screen with just that value than a screen that includes that data along with a lot of additional, superfluous information. Separating the grain of edible data from the screen full of chaff can be painful. It also

implies that a Web page cluttered with advertisements is not as simple to use as one without — a moral likely lost on advertisement-supported Web page designers — and that there's no point in getting details about the purchase of a product you can't sell. Raskin's book, *The Human Interface*, contains a wonderful illustration of these points, showing how attention to the user's task and the costs of different activities can be combined to produce more usable systems.<sup>4</sup>

### Dimensions of Choice

Interface design is complicated, both in the complexity of the design space and the pragmatics of usable systems. The user-interface designer can choose from a large variety of mechanisms, including (to name a few) command languages, character languages, natural languages, direct-manipulation graphics, forms, and menus. We can indicate our choices to systems via switches, dials, keyboards, mice, tablets, joysticks, touchpads, and touchscreens — or perhaps by pushing foot pedals, speaking, whistling, or grunting rhythmically. The machine can respond with a display that ranges from lighting a single point to presenting a high-resolution, color animation. It can simultaneously squeak, flash, and speak. For blind programmers, there is even a device that converts ASCII to Braille, invoking the sense of touch. (For the moment, proprioception has been employed only in certain flight simulators and games, and designers have made little use of the sense of smell — surely serious oversights.) These controls can be organized into command sequences, dialogs, macros, forms, menus, direct-manipulation systems, and so forth.

### Complexity Is in the Mind of the Beholder

The simplicity of an artifact varies depending on who is doing the evaluation. My favorite example of this comes from cooking.

"When Carl Sonthemimer launched Cuisinart in this country more than a decade ago, he had to do a bit of everything at the fledgling enterprise, including writing the instruction manuals for the newfangled food processor. So the MIT engineering PhD was particularly flattered when he received two letters from customers praising the depth and thoroughness of his instructions...But 'when I called up to thank them, I found out they were (also) engineering PhDs from MIT.'"<sup>5</sup>

This begs the question of what we want from a good user interface? Many conflicting things.

### Reduced Mental Load

A user interface's primary goal is to reduce the mental load (the cognitive overload) on the user. People have trouble remembering things, solving problems, and planning and expressing goals. To the extent that a device makes them do these things, they suffer. To the extent that a device saves them these mental actions, they appreciate it. Thus, systems reduce the user's mental load when designed to achieve goals that match the user's – systems that present a coherent and comprehensible picture of their internal behavior, that understand the natural, conversational idiom of the user's task, and that understand, if not the natural task idiom, at least the natural idiom of the device. (This goal excludes, of course, artifacts designed precisely to exercise one or more of these factors – games, puzzles, and teaching machines come to mind.) In general, interfaces mediate complexity. To the extent that the interface reduces a device's complexity (without subverting its utility) it will be successful.

### Accommodating Individual Differences

People are not all the same. Good interfaces recognize this fact and adjust (or at least can be adjusted) for such differences.

**Goals.** People differ in what they're trying to accomplish. Often, these differences are driven by using the same system to perform different jobs. For example, the ideal word processor for someone composing a document is not ideal for someone editing a document someone else has written. (Naïvely, the composer needs to easily insert and move text and to obtain an overview of the document.

The editor is likely to be processing the document serially and to be more concerned with search and localized alteration than insertion and overview. The editor's changes ought to be marked; tracks of the author's changes are likely to get in the way.) The receptionist interacts with the email system almost exclusively by sending short messages to specific individuals; the manager likely reads more (and longer) emails, sends mes-

sages to groups, and files and organizes messages.

**Task expertise.** Users differ in their knowledge of the task at hand. Good interfaces give guidance to task novices; poor interfaces obstruct experts with unnecessary kibitzing.

**System expertise.** Similarly, users differ in their knowledge of the device they're using. Again, an ideal interface teaches as much of the system as desired to novices, while not interfering with experts' actions.

**User preferences.** Some people prefer to do things a certain way. On one hand, it's better if the machine can do it their way, rather than forcing users to think its way. For example, I remember a program editor that required the user to specify an opera-

## Classified Advertising

**SUBMISSION DETAILS:** Rates are \$110.00 per column inch (\$125 minimum). Eight lines per column inch and average five typeset words per line. Send copy at least one month prior to publication date to: Marian Anderson, Classified Advertising, *IEEE Internet Computing Magazine*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; (714) 821-8380; fax (714) 821-4010. Email: [manderson@computer.org](mailto:manderson@computer.org).

**COORDINATOR, COMPUTER APPLICATIONS, Tampa, Florida.** Administers network management servers running Linux and Solaris. Develops monitoring and management applications using scripting languages and C in a Unix/GNU development environment. Tests and evaluates hardware and software to determine efficiency, reliability and compatibility using equipment such as computers, network simulators and protocol analyzers. Tests, evaluates and documents hardware or software for purchase. Designs networks based on technologies such as fiber optics. Analyzes fiber optic transmission systems. Reads technical manuals and brochures, communicates with manufacturers' and vendors' engineers to determine equipment that meets established

standards and users needs. Compiles and presents budgets for recommended solutions. Conducts surveys to determine user needs. Trains users in use and maintenance of equipment. Identifies and solves data communications problems and helps users to identify and solve problems. Writes technical specifications to send to vendors for bid. Oversees or assists in installation of communications hardware and software. Performs minor equipment repairs. Performs webmaster duties. **REQUIREMENTS:** Master's Degree in Physical Sciences or Computer Science or the equivalent. Salary: \$43,036/yr. Hours: 40 hr/week - 8:00 am to 5:00 pm. **CONTACT:** All interest applicants should contact: Agency for Workforce Innovation, PO Box 10869, Tallahassee, FL 32302. Job Order #FL-2528350.

tion's arguments before the operation itself (that is, "tell me what you want to insert, and then tell me where to insert it"). I found this difficult because I usually decide I want to do an insertion before I've composed what it is I'm going to add. Forced to compose first, I forget where it was going to go and what I was going to do with it there. However, I know at least one other person who assured me that he couldn't imagine editing any other way.

On the other hand, it has been argued that individual customization is a barrier to interpersonal interaction – with customization, you can't work effectively on a device with another person, easily guide them through debugging a problem, or walk up to another system and have it behave in a predictable fashion.

**Mental differences.** Not everyone thinks like a computer-science geek. Systems should therefore present and accept information in the manner that matches the user's mental orientation.

**Accommodating People**

Just as people have individual differences, there are also some human universals. I often feel that the WIMP (windows, icons, menus, and pointing device) interface style has been optimized for people with three hands. Similarly, to err is human, though too many systems deal poorly with allowing users to recover from their mistakes.

**Producing a Positive Effect**

The user should feel good about interacting with the system. This includes augmenting positive feelings such as curiosity and interactivity, and reducing negative feelings such as frustration and boredom.

**Promises and Reality**

Acting in opposition to good interface design is the reality of engineering projects. Good interfaces are expensive. They require extra effort to design and develop and place greater demands on the underlying hardware than naïve interfaces. Quality is often refined only

through extensive user testing. A similar conflict rages within the interface for any given system – the desire to expand functionality can conflict with reducing complexity just as assigning a particular action to a given command or key limits the use of that command or key for other actions.

Next issue, I'll build on this brief sketch of the goals of good interface design. We'll consider in more detail the elements of the problem space that interface designers should recognize in doing their work. □

**References**

1. E.R. Tufte, *The Visual Display of Quantitative Information*, Graphics Press, 1983.
2. E.R. Tufte, *Visual Explanations: Images and Quantities, Evidence and Narrative*, Graphics Press, 1997.
3. E.R. Tufte, *Envisioning Information*, Graphics Press, 1990.
4. J. Raskin, *The Human Interface: New Directions for Designing Interactive Systems*, Addison-Wesley, 2000.
5. "The Nightmare of Product Instructions," *San Jose Mercury News*, 26 Dec. 1986, p. 20E.

**IEEE INTERNET COMPUTING**

IEEE Computer Society Publications Office  
10662 Los Vaqueros Circle  
Los Alamitos, CA 90720

**EDITOR IN CHIEF**

Robert E. Filman • filman@computer.org

**ASSOCIATE EDITOR IN CHIEF**

Li Gong • li.gong@sun.com

**EDITORIAL BOARD**

Jean Bacon • jean.bacon@cl.cam.ac.uk  
Elisa Bertino • bertino@dsi.unimi.it  
Scott Bradner • sob@harvard.edu  
Siobhán Clarke • siobhan.clarke@cs.tcd.ie  
Fred Douglass • f.douglass@computer.org  
Stuart I. Feldman • sif@us.ibm.com  
Ian Foster • foster@cs.uchicago.edu  
Monika Henzinger • monika@google.com  
Michael N. Huhns • huhns@sc.edu  
Leonard Kleinrock • lk@cs.ucla.edu  
Doug Lea • dl@cs.oswego.edu  
Frank Maurer • maurer@cpsc.ucalgary.ca  
Daniel A. Menascé • menasce@cs.gmu.edu  
Chris Metz • chmetz@cisco.com

Charles J. Petrie • petrie@nrc.stanford.edu  
(EIC emeritus)

Krithi Ramamritham • krithi@cse.iitb.ac.in

Munindar P. Singh • singh@ncsu.edu

(EIC emeritus)

Craig Thompson • cwt@uark.edu

Steve Vinoski • vinoski@ieee.org

Dan S. Wallach • dwallach@cs.rice.edu

Jim Whitehead • ejw@soe.uscs.edu

**IEEE Communications Society Liaison**

G.S. Kuo • gskuo@ieee.nccu.edu.tw

**STAFF**

Lead Editor: Steve Woods

swoods@computer.org

Group Managing Editor: Gene Smarte

Staff Editors: Scott L. Andresen,

Kathy Clark-Fisher, and Jenny Ferrero

Production Editor: Monette Velasco

Magazine Assistant: Hazel Kosky

internet@computer.org

Graphic Artist: Alex Torres

Contributing Editors: Cheryl Baltes,  
David Clark, Greg Goth, Keri Schreiner,  
and Joan Taylor

Publisher: Angela Burgess

Assistant Publisher: Dick Price

Membership/Circulation Marketing

Manager: Georgann Carter

Business Development Manager:

Sandy Brown

Advertising Supervisor: Marian Anderson

**CS Magazine Operations Committee**

Bill Schilit (chair), Jean Bacon, Pradip Bose,

Doris L. Carver, George Cybenko, John C.

Dill, Frank E. Ferrante, Robert E. Filman,

Forouzan Golshani, David Alan Grier,

Rajesh Gupta, Warren Harrison,

Mahadev Satyanarayanan,

Nigel Shadbolt, Francis Sullivan

**CS Publications Board**

Michael R. Williams (chair), Michael Blaha,

Mark Christensen, Sorel Reisman,

Jon Rokne, Bill Schilit, Linda Shafer,

Steven L. Tanimoto, Anand Tripathi

