Mezzanotte, R. J., and Kibler, C. T. (1984). A comparison of three SQL-based query products. Endicott Human Factors, IBM Confidential.

Neal, A. S., Simons, R. M. (1983). Playback: A method for evaluating the usability of software and its documentation. San Jose California: IBM Human Factors Center, Technical Report HFC-48.

Ogden, W. C., Korenshtein, R., and Smelcer, J. B. (1986). An intelligent front-end for SQL: Correcting user errors with Natural Language menus. San Jose, California: IBM Human Factors Center, Technical Report HFC-55, Internal Use Only.

Williams, M. D. (1984). What makes RABBIT run? International Journal of Man-Machine Studies, 21, 333-352.

Lola M. Arnold BS, MA is a recent graduate from San Jose State University, San Jose, California. She is currently contracting with Apple Computer as a human factors specialist in Software Quality Assurance. She can be reached at her home address: 325 Lakeview Drive, Felton, CA 95018 (408) 335-3594.

CREATING CONSISTENCY IN THE USER INTERFACE: OPINIONS AND PROCEDURES OF SOFTWARE DEVELOPMENTS EXPERTS[1]

ALAN J. HAPP, PH.D., KAREN C. COHEN, PH.D.

INTRODUCTION

The User Interface Technology department of IBM, Boca Raton, recently commissioned a study on the topic of consistency in the user interface. Of interest were general principles related to consistency, the methods of developing and/or attaining it in product development, and the relative effectiveness of IBM's Systems Application Architecture manual, *Common User Access - Panel Design and Interaction (CUA V 1.0 1987)*, in helping independent software vendors achieve consistency in their products.

The findings reported here relate to issues of physical, dialog, and conceptual consistency in the user interface, with an emphasis on the software design and development process. The focus was on guidelines and elements in IBM's common user access--panel design and interaction manual developed by it's Systems Application and Architecture Group. However, the approach was generic and the findings apply to a variety of products and processes.

Two pools of interviewees were involved: (1) seven independent software vendors (ISVs) to IBM, and (2) eight non-IBM experienced and expert software developers (NSDs).The findings tend to be congruent in both pools, emphasizing the importance of physical, conceptual and dialog consistency in products. Specifically, the use of tools and examples to be worked and the clear statement of a CUA style were seen as the most efficient and expedient way to attain consistent interfaces.

Themes and Observations

The major topics running through the entire set of interviews could be classified as follows:

- Programming requirements
- Design principles
- Value of CUA style guide/potential of style guide
- Tools
- Rigidity and Flexibility
- Recommendations for implementing a CUA process

More specifically, the interviewees recommended the following concerning these general themes:

Programming Requirements

If a single recommendation is implemented from this study, it should be that IBM develop and distribute TOOLS that not only create but enforce consistency in the products to be developed. Tools allow programmers to use their time more creatively for the content of the products they are developing. Second to tools (which most of the vendors developed idiosyncratically but anyway for their specific products) were "examples to be worked," i.e., tools that allow for adaptation of real code as appropriate for the application's specific needs.

Design Principles

The ISV's commented in this area primarily in terms of "rules of thumb" for assessing consistency; the NSD more in terms of guidelines to be followed. ISV concerns were, primarily, with physical consistency, and they used approaches to be sure that a product could be recognizable as "belonging to a family." For example, one vendor cited the "Five Foot Rule": i.e., from a distance of five feet do the screens appear to be consistent. Exactness at a more detailed or closer level was not necessary or even seen as desirable.

The NSD group had a broader set of principles or suggestions, oriented more toward creating consistency "a priori" than in a working product. Some of their strongest recommendations were that:

- There should be a one-to-one mapping of use to function and no other way to do things. (This did not square with later recommendations of allowing individuals to set their own style. In this instance the concern was with dialog rather than physical consistency).
- NEVER block or obscure the problem with a window. It makes it harder for users to remember and deal with it.
- Do not have redundant information on the screen (e.g., ESC appearing 3 times).
- Have one and only one meaning for each option.
- USERS should be the focus of the level of consistency in a product. Consistency to the programmer may not be

---

1 © IBM Corp., 1989.

consistency to the designer, and consistency to the user may differ from both.

- If there is more than one way to get things done (e.g., mouse and function keys,) let user determine what fits with his/her own preference and style.
- The system should never destruct without permission and confirmation.
- Develop error messages "backwards," from problems users encounter in the real world. Programmers and developers pre-thinking error and help messages do not always match user requirements or needs.
- Screen design and functional areas employed have a strong relationship to consistency in terms of "feel" regardless of looks.

Value of a CUA Style Guide Potential

The primary value of the CUA Style Guide and the potential to the developer/vendor (beyond development) was in marketing. All interviewees opted or preferred to have a guide--which many referred to as a "Bible." They saw a CUA conforming product as very important to marketing and as an aid to developing future products, not just a document of what can already be done. Guides create a market for integrated packages and are very much wanted by developers. Many interviewees said they prefer to follow the guidelines even if there may be a better way to do it for the reasons stated above.

Tools

The value of tools cannot be overstated. They are viewed as the key to productivity, consistency, and speed. You must have them if you are serious about achieving consistency. Tools should not be 100% fixed, but should allow the programmer to be creative in emerging areas (e.g., alternative-media Help) and application specifics. Tools can dramatically cut the cost of development.

Rigidity and Flexibility

Both the ISVs and NSDs felt that rigidity at the surface was mandatory and that dialog should be absolutely inflexible for the user. Less than 100% control is possible, but close to 100% should be the goal. There should, however, be flexibility at the "meat" of the application. Variations should be allowed if it is better for the application or if the user can do a task in one step rather than five (required by the standards).

The ISVs mentioned many specific approaches for consistency changes or revisions in the CUA guidelines--e.g., needing to allow menu trees deeper than two levels, standard menu positions and standard calls for actions such as file, open, new, close, save, save as, edit, copy, clear, cut, undo, top, etc.

CUA Process--How do you Implement it?

In general, the ISVs used general principles from external interface guidelines , a "subset" version or their own version of the CUA guidelines, and a conscious striving for guidelines or styles in the user interface interactions. They had internal standards, more structured than the IBM CUA guidelines, which they followed.

The NSD group had general recommendations: tools, one person in charge for quality control, storyboarding, continual dialogue with end-users (formative evaluation) and test, and review of their products at all levels.

Summary

For this study of consistency in the user interface, we interviewed 20 experienced software developers individually and/or in small groups with a set of questions determined by the UIT. The interviewees fell into two groups: (1) IBM ISVs and (2) NSDs, experienced individuals involved with the same problems of software development at other companies or institutions.

The results of all interviews indicate a high degree of consistency among their responses. The strongest recommendations concerning the CUA Guide were that it should be modified to 1) meet the needs of their users in different environments (most of the IBM-ISVs rewrote the document to make it usable by their programmers); and 2) provide more information about the style of CUA rather than the details (which should be handled by a tool).

Their strongest general recommendations were that IBM provide "tools" or "examples to be worked" which would not only create and enforce consistency in products but also save enormous costs in product development. Another ISV suggestion was that IBM identify "consistent" software with a seal of approval, etc, which would help in marketing. Other suggestions are covered in the body of the report in detail.

Recommendations for Action

In terms of the range of reactions and the quality of the feedback this study uncovered, it is encouraging to note that several actions suggested by the interviewees are being considered for future releases of the CUA. Some of these are:

- A separate text for independent workstations and non-programmable terminals. To ensure consistency, both efforts must be pursued.
- A style guide is in preparation with increased attention to graphic design.

The results of the study also suggest some new directions. The vendors' requests for consistency across products could be addressed with actions in the following areas:

- Increased communication between CUA and Vendors. Feedback, such as from this study, and meetings with groups of vendors to provide CUA direction are important steps in understanding our customers needs. Discovering what our customers' customers need can help us build better tools and, ultimately, natural-to-use interfaces.
- A rigorous design review of each product. The review could be modelled after the design review program which is part of the Corporate Design Program.
- There is a need to address graphic issues and interaction issues at a corporate level making representatives from product development accountable for the direction of the product.
- Continued work with vendors as part of competitive analysis. The vendors had much to say about the

approach to consistency taken by other vendors such as Apple and DEC.

The final recommended action is to continue to seek vendors' feedback regarding consistency. The vendors realize that consistency can play a part in helping their customers reduce training and support costs. They appreciate IBM's willingness to seek their feedback regarding consistency issues. The effort can be included (at little cost) as part of the early ship program and through continuing work with ISVs.

About the Authors

Dr. Alan Happ works as an Advisory Human Factors Engineer in the Design Center at the Entry System Division of IBM US in Boca Raton, Florida. He joined IBM in 1982 and has worked on human computer interaction issues for personal and mid-range business systems. His research interests focus on the human factors required for the design and development of products that meet the users needs. He received M.A. and Ph.D. degrees in Experimental Psychology from Miami University, Ohio.

Dr. Karen C. Cohen has been at MIT for 15 years. She is currently Principal Research Associate at the Center for Cognitive Science and Project Athena. Known primarily for her work in academic and industry on evaluating the impact of computers and learning, she has published eight books and over 100 articles and monographs in the field. She has consulted to IBM on several projects. She received her B.A., Magna Cum Laude, in Social Relations from Harvard University and her M.A. and Ph.D. in Social Relations and Educational Psychology at The John Hopkins University.

The study was sponsored by the User Interface Technology Dept., Entry Systems Division, IBM U.S., Boca Raton, Florida. Special thanks are extended to H. Dulaney, manager of UIT, and Doretha Lippett, lead for the task force on consistency.

AN EMPIRICAL APPROACH TO THE EVALUATION OF ICONS

JAYSON M. WEBB, PAUL F. SORENSON, NIC P. LYONS

General Abstract

This poster provides a definition and taxonomy for iconic communication and describes the use of formal psychological tools and methods in the evaluation of icons. The methods that can be usefully applied include:

1. Psychophysics
2. Scaling
3. Recognition/Memory Testing
4. Statistical Modeling / Analysis

Examples of some of these approaches are provided from pilot studies currently under way at HP. Analyses used include Multi-Dimensional Scaling (MDS) and Cluster analysis. Results can be applied to development of metrics, standard methods, and design guidelines.

Detailed Abstract

Definition and Taxonomy

An icon is a pictogram which can be selected or otherwise interacted with by the user of a system interface, and which represents one or more of the following:

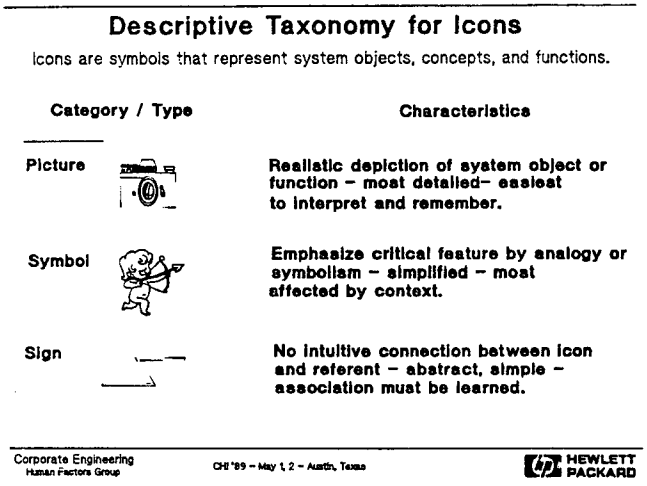
The functions of the computer system, The system objects upon which these functions act, Certain types of system status.

The user interacts with Icons in several ways, including: Selecting (Activation using mouse, or other input device), Moving, Copying, and Deleting.

Types of Icons

There are 3 types of icon (see 1st Figure), each of which conveys its meaning in a different way:

Figure 1



Pictorial: Realistic depiction of system object or function. Reference by resemblance. Have the most detail, are the most concrete, easiest to interpret and remember.

Symbolic: Depicts a critical feature of the referent object or function through analogy or symbolism. Reference by symbolism. Representation is simplified - most affected by context of presentation (e.g., system metaphor employed).

Sign: No inherent, intuitive connection exists between the icon and its referent. Relationship between icon and system object or function must be learned by rote. Reference by learned association. Simplest, and most abstract.

Interactive Attributes

1. Detectability (in a crowd, distinguishability)
2. Legibility
3. Interpretability