# HUMAN FACTORS APPROACHES TO PROTOTYPING AND EVALUATING USER INTERFACES

**Nicholas I. Benimoff and William B. Whitten, II**

***Nicholas I. Benimoff*** *is a member of technical staff and* ***William B. Whitten, II****, is supervisor in the User Interface and Quality Planning Department at AT&T Bell Laboratories in Holmdel, New Jersey. Mr. Benimoff is responsible for user interface design and evaluation of AT&T voice store-and-forward services. He received an A.B. from Clark University and an M.S. and Ph.D. from Columbia University, all in psychology. He joined AT&T in 1985. Mr. Whitten is responsible for designing user interfaces for transmission, data, and attendant services; promoting state-of-the-art user-interface technologies; and designing multimedia knowledge systems. He received an A.B. from Indiana University and an M.S. and Ph.D. from the University of Michigan, all in psychology. He joined AT&T in 1981.*

Human factors contributions to the design of user interfaces are likely to have the greatest impact at early design stages. Rapid prototyping therefore provides an important vehicle for incorporating quality and usability from the outset. Through realistic experience and interaction with the proposed user interface, users and user-interface designers can work together to detect and correct problems before a major development investment is made. The prototyping approach is especially powerful because it provides the opportunity for (1) early user feedback and performance data, (2) efficient, focused communication with users and developers, and (3) iterative design and test. In this paper we illustrate the prototyping and evaluating process with three examples of AT&T products and services.

## Introduction

Over the last few years, traditional systems development methods have been replaced by a more dynamic approach in the computer and information systems industries. This approach—*prototyping*—is typified by building a system model, prior to formal development of such a system, that looks and behaves as the system will look and behave once development is complete. The prototype offers an early and realistic view of the system so that modifications and improvements can be incorporated into the final product or service.

The prototyping approach is generally useful for clarifying technical feature requirements and for estimating development efforts. The approach has been used effectively by a wide variety of engineers in all phases of the product and service development processes. This paper will focus on the use of prototyping by human factors specialists, since the approach is especially powerful in designing user interfaces that meet customer needs.[1-3] A prototype developed by the human factors specialist serves as a *user interface demonstration system* (UIDS). The UIDS can be created in sufficient detail to give users a realistic preview of the product or service user interface. Users' interactions with the UIDS provide behavioral data and other feedback that guide improvements to the user interface, ensuring a high-quality and successful product or service.

**Benefits of the Prototyping Approach.** The design of an efficient and pleasant user interface, via prototyping, is an evolutionary process that is affected greatly by project team and user feedback. The lessons learned at each test iteration are incorporated into the next design iteration. This approach provides these important benefits:

1. Valuable user feedback is obtained *early in the design process.*
2. Qualitative and quantitative human performance data can be collected, usually within the context of the specific application.[4] Indices measured include learning time, task completion time, the number and types of errors made, and user evaluations (e.g., preferences and satisfaction) of various aspects of the system.
3. More efficient and quicker communication with developers and users occurs, often leading to savings in time, resources, and money. The prototype provides a common point for user-interface designer-developer communication so that problems can be identified early in the development process. Boehm found that failures detected at, for example, the delivery stage are as much as 50 times more costly to correct that those found at the system-requirements and high-level design stages.[5]
4. Quick modification for iterative design and test is encouraged,[6,7] providing a potent opportunity for an improved user interface.[3,4]
5. Prototyping delivers measurable savings in the areas of project team members' effort, resource allocation, time, and, of course, money.[8] In fact, the literature suggests that prototyping can lead to a reduction in development cost and time by a factor of at least 4 or 5.[9,10]

Overall, prototyping is critical in the design and development of user interfaces that are of high technical quality. In addition, it provides a healthy opportunity for increased teamwork, both among project team members and between the project team and potential users.

**Prototyping Methodology.** Prototyping stresses interaction among and communication between project team members (including potential system users—see Figure 1a) in the belief that this leads to the best possible product or service. This approach may be contrasted with the more traditional system development life cycle (shown in Figure 1b). Note that the former model is an interactive and iterative one, where design decisions are based on input and feedback from a variety of critical sources, while the latter model is better characterized as a serial one, where there is less opportunity to take advantage of the benefits of the interactive process.

Prototyping involves a number of steps. The human factors specialist must:

1. Learn what functions the product or service will provide
2. Work with the project team and with potential users to define clearly the detailed capabilities of the product or service
3. Develop a model of the user interface that simulates the appearance and behavior of key elements of the product or service, using knowledge of human factors, cognitive psychology, and human behavior
4. Demonstrate the prototype to project team members and potential users of the system to collect comments and test for usability.

These are the *essential* elements of prototyping. In addition, much is gained by the iterative process where opinions from members of the project team and from potential users, as well as objective data from empirical usability experiments, lead to successive modifications of the prototype.[11-14] Until recently, design iterations have been costly and time-consuming. The development of powerful tools that allow quick modification of the prototype has changed that. The most dramatic recent improvements have been in the domain of color graphics for interactive computing. These software

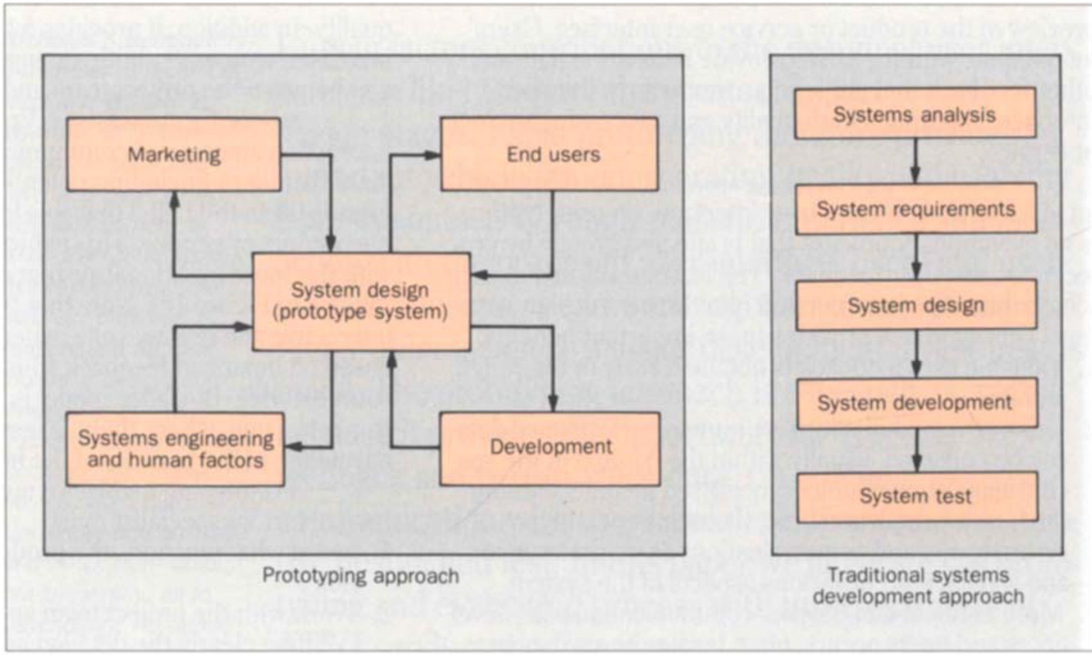


**Figure 1. How the prototyping approach to systems development contrasts with the traditional systems development approach.**



toolkits allow the user-interface designer to specify the appearance of screens and displays in a prototype as well as the flow of control among screens and displays contingent on user input (see Reference 15 for a survey of selected graphical prototyping tools).

### Examples of the Prototyping Approach

Within the User Interface and Quality Planning Department at AT&T Bell Laboratories, a large number of projects have benefited from the integration of prototyping into the development process. These include the user interface designs for several Advanced 800 services, Accunet® T1.5 information manager, and the Accumaster™ integrator. Within all of AT&T, an even wider variety of products and services have been designed via rapid prototyping. In this section we will highlight three examples of how human factors specialists at AT&T Bell Laboratories and AT&T Consumer Products Laboratories used the prototyping approach.

**Prototyping a Network Management System.** AT&T's *advanced services quick implementation capability* (ASQIC) process is an example that illustrates how high-quality user interfaces are developed efficiently by prototyping. The most recent application of this process is the *integrated network controller* (INC), which was given to one AT&T customer on a trial basis. It made possible end-to-end network management for Dataphone® data communications service by providing features such as service restoration, performance management, and fault management. The INC system user interface employed interactive color-graphics technology to provide a workstation display of network information. The user interface included a mouse-controlled cursor, multiple overlapping windows, and user-system dialogue devices such as menus, windows, and "soft" (i.e., on-screen) command buttons.

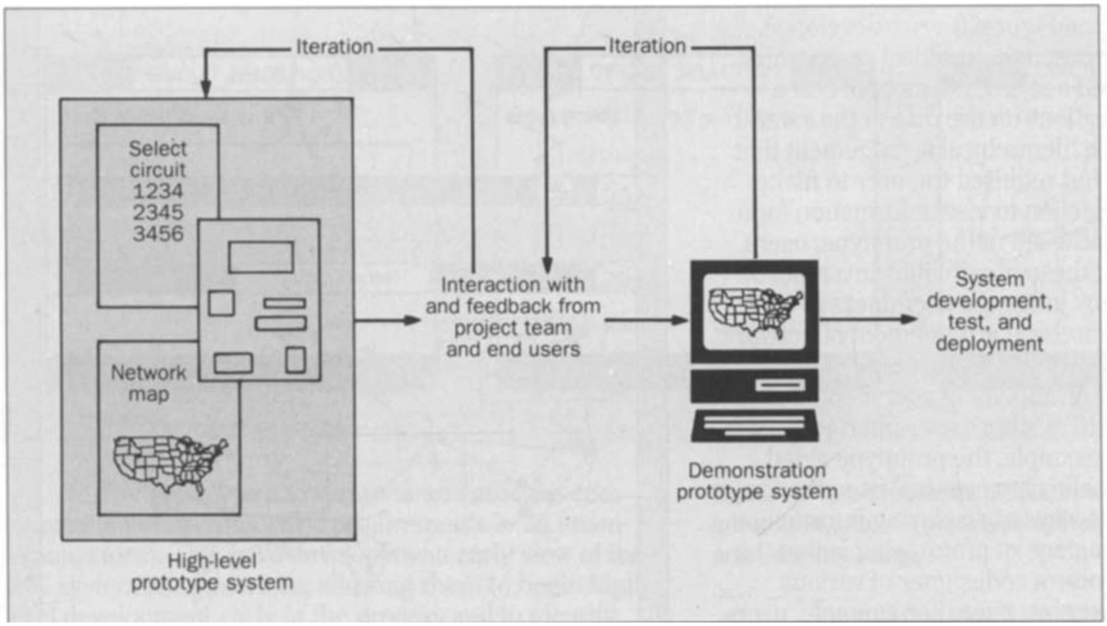


**Figure 2. The process associated with the prototyping approach.**



One of the first formal steps in the evolution of the INC project was the early development of a user-interface prototype system (see Figure 2). The prototype system was at first a high-level one that stressed concepts to be incorporated into the user interface. It was composed of paper-and-pencil sketches, and later, a series of static, "slide-show" type graphic screen displays. These displays were used to elicit comments from project team members regarding the overall view of the INC system user interface. However, little could be learned about three key attributes: procedural rules, degree of interaction, and usability. To address these issues, a more sophisticated and realistic prototype was necessary, one that would be interactive and similar in most respects to that of a completed service's user interface. This need was filled by using a series of commercially available tools (V. I. Corporation's DataViews® software, including both DVDraw® and DVTools® software, and Hewlett-Packard's HP® Windows). These were augmented by software developed within the Planning Systems Department of AT&T Bell Laboratories, running on a Hewlett-Packard 9000/350® color-graphics workstation. This combination allowed the user-interface designer to create screens and link them into a demonstrable prototype. These screens were easy to update in response to feedback from the project team and users. Thus, prototyping allowed for a series of early and continuing views of the service, usable by marketing, systems engineering, development, and the user.

The user had several opportunities *prior to and during development* to affect aspects of the INC system (e.g., technical features and types of information gathered and displayed). With respect to the user interface, this included the form of information presentation as well as system dynamics, in some cases. For example, working with the prototype system gave users the opportunity to discuss how their data telecommunications network should be represented. Several examples of different network map displays (see Figure 3) were developed. Some displayed the network as a graphical representation, while others showed a stylized depiction. Some displayed many user locations on the map at the same time, while others used a hierarchical arrangement that reduced screen clutter, but required the user to make more than one menu selection to view information for a particular location. With the aid of the prototype, users were able to test each of these possibilities in a realistic context and could thereby judge the usefulness of each format. A particular hierarchical arrangement of network elements was found to be the easiest to read and the quickest to use, and this arrangement was adopted as a key component of the INC system's user interface.

In the previous example, the prototype aided user-interface design by allowing eventual users to compare and tests alternative ways of displaying information. In contrast, another advantage of prototyping is that the user can essentially become a codesigner of various pieces of the system's user interface. For example, users sat down with the user-interface designer and jointly designed a system for circuit identification (e.g., for testing or report generation) that was understandable and easy to use. Among the possibilities considered were:

1. Limiting the users to text entry of circuit identifiers
2. Listing circuit identifiers and allowing users to select one with the mouse
3. Presenting a graphical representation of the circuits from which the user could make a selection (see Figure 4).

Each design was discussed with users, and more importantly, each was built into versions of the prototype so that the users could experience the feature within the context of the whole system. In this way, the prototyping technology allowed those who would eventually be customers to help design the system.

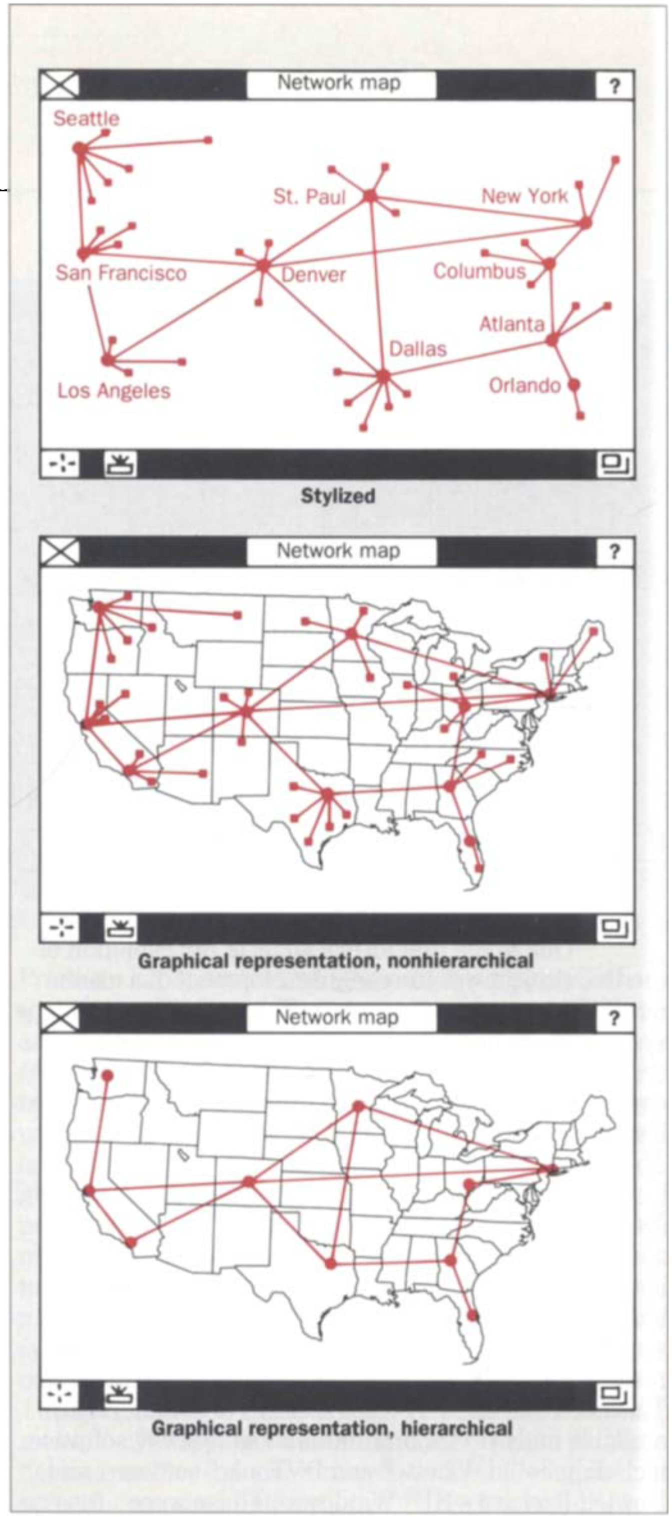


**Figure 3. Three methods of network map display which were integrated into the prototype system for user preference and usability study.**

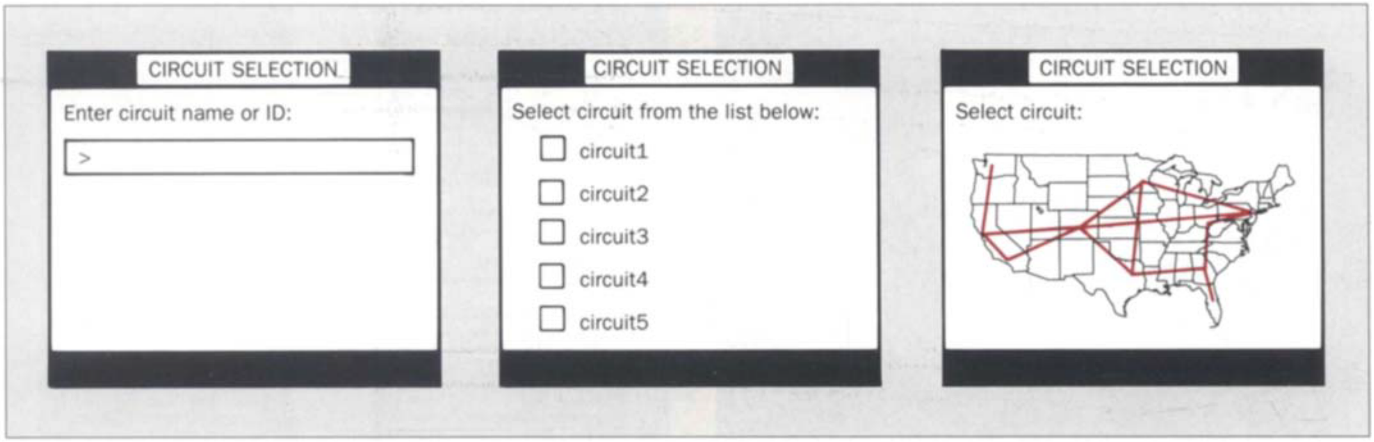


**Figure 4. Three methods of circuit selection which were integrated into the prototype system for user preference and and usability study.**

The prototype allowed swift and efficient communication of user-interface requirements in an unambiguous form. This gave developers an early view of the INC system user interface, allowing them to begin high-level development early in the process and to identify features that were too costly to develop. If a picture is worth a thousand words, and if a prototype is composed of many separate pictures, then it is clear that a well-developed prototype can be much more efficient than written requirements as a communications vehicle.

Prototyping of this graphical user interface, in particular iterative rapid prototyping, allowed early and continued user input and resulted in a telecommunications management system that met the customer's needs. The customer's positive opinion of the user interface was shown by high satisfaction and importance rankings for the INC system's six major features.

**Prototyping User Interfaces for AT&T Products.** The previous example illustrated the use of rapid prototyping techniques to optimize color-graphics software user interfaces. Similar techniques have been useful in the design of hardware devices. For example, the prototyping approach was used to evaluate two alternative user-interface designs of the AT&T HT series cordless telephone product line.

A computer-based prototype was developed to simulate the operation of a cordless telephone handset (see Figure 5). By studying the way potential users interacted with the prototype, issues relating to button labeling, button layout, and operational procedures could be resolved. The prototype was implemented on a high-resolution graphical workstation using a touch-sensitive screen. Custom software was written for a DEC MINC-11 system with an Elographics touch screen.

A major objective of the study was to assess users' performance on cordless telephones that have only button controls rather than a mixture of slide switches and buttons. Software-controlled buttons were portrayed on a touch-sensitive screen at a significant cost savings compared to building physical prototypes. For this study, two prototypes of the handset were developed, each with a different user interface for its procedural operations and button layouts.

Operational procedures for the first prototype were developed through experience with an existing AT&T product design, through incorporation of

**Figure 5. Computer-based prototype simulates the operation of a cordless handset.**

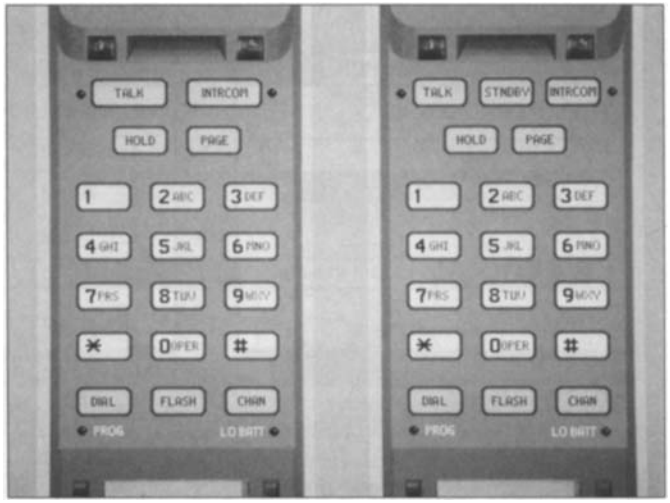


**Figure 6. Simulated cordless telephone faceplates for prototype I (right) and prototype II (left).**

additional functionality, and through studies of competitive products. More specifically, the handset incorporated three major function buttons: *Talk*, used to go off-hook; *Standby*, used to go on-hook; and *Intercom*, used to initiate an intercom conversation between the base unit and the handset.

The second prototype design included only two major function buttons, *Talk* and *Intercom*. (Figure 6 shows both faceplates.) Two major differences were introduced in the second prototype: (1) *Talk* was a toggle operation in which pressing the button once took the telephone off-hook and pressing it again put the telephone back on-hook and (2) a user hung up after an intercom conversation by pressing *Intercom* a second time, instead of pressing *Standby* as in the first prototype. Identical methodologies were used for both studies so that results could be compared.

Participants in the experiment were given a series of tasks and asked to use the buttons on the touch screen as they would a telephone. Table I lists the nine types of tasks and the correct responses for each of the prototypes. Each participant received a total of 44 trials in one of three orders. They were provided with auditory and visual feedback as a result of correct actions. For example, when the *Talk* button was pressed, a speaker emitted dial tone and associated light-emitting diodes (LEDs) lit on the touch screen. The prototypes indicated incorrect actions by a written error message on the screen; they did not allow users to traverse a full error path. Number of attempts, time per task, and error types were recorded to evaluate performance. Users were also asked to rate the difficulty of each task. Comments made during the studies were noted.

Table II presents error data from the two prototypes. Performance, as measured by number of errors,

**Table I. Types of Tasks Used in HT Series Prototype Study**

| Task | Correct response | |
|---|---|---|
| | Prototype I | Prototype II |
| Make a call | *Talk*-7-8-1-5 | *Talk*-7-8-1-5 |
| Hang up call | *Stndby* | *Talk* |
| Answer incoming call | *Talk* | *Talk* |
| Page base unit | *Page* | *Page* |
| Initiate intercom conversation | *Intrcom* | *Intrcom* |
| Hang up intercom call | *Stndby* | *Intrcom* |
| Put call on hold | *Hold* or *Intrcom* | *Hold* or *Intrcom* |
| Return to held call | *Talk* | *Talk* or *Intrcom-Talk* |
| Without hanging up call, initiate intercom conversation | *Intrcom* or *Hold-Intrcom* | *Intrcom* or *Hold-Intrcom* |

**Table II. Errors Committed as Percent of Total Trials per Task**

| Task | Prototype I (30 participants), percent | Prototype II (28 participants), percent |
|---|---|---|
| Make a call | 52.4 | 43.8 |
| Hang up call | 5.8 | 13.4 |
| Answer call | 12.2 | 2.9 |
| Page base unit | 17.8 | 8.2 |
| Initiate intercom | 8.6 | 9.2 |
| Hang up intercom call | 7.7 | 7.4 |
| Put call on hold | 3.2 | 0 |
| Return to held call | 41.3 | 26.8 |
| Without hanging up, initiate intercom | 1.6 | 3.45 |



did not differ dramatically. On the other hand, evaluating the types of errors committed with both prototypes yielded two important results:

- The most frequent error was forgetting to press the appropriate button to get dial tone.
- The radio and cordless industry-standard label *Standby* was confusing to telephone users, and the alternative tested in the second prototype, *Talk*, was no better.

While performance on some tasks improved with the second prototype, a few more errors associated with hanging up were committed. Designers later evaluated alternative button labels for the product. As a result of further human factors experimentation, the *Talk* and *Standby* labels were replaced with the less confusing alternatives *Phone* and *Off*, respectively.

In summary, using a computer-based prototype greatly enhanced the quality of the user interface by providing a mechanism for iteratively evaluating alternative designs and to prioritizing identified human interface

**Table III. Responses to Cueing Methods**

| Cueing method | Percent appropriate response (120 participants) |
|---|---|
| Initial prompt—menu—"now" | 65 |
| Initial prompt—menu—tone | 58 |
| "At the tone"—initial prompt—menu—tone | 79 |

issues. It also reduced cost, since it is less expensive to develop a computer-based prototype than an actual hardware prototype. Finally, the prototype allowed the development team to function more efficiently and to have more confidence in its design decisions.

**Simulating an Automatic Operator Service.** The integrated network controller and the AT&T HT series cordless telephone examples highlight the use of graphical software prototypes in the design of high-quality user interfaces. However, prototypes can be much more complex, as in the example presented below. Here the prototype included telephone equipment and a network simulator, as well as live attendants simulating automatic speech recognition equipment, all intended to address certain issues in the design of the user interface for an AT&T operator service.

A new operator services position is being developed that can deliver special-billing (e.g., collect, credit card, or billed to a third party) calls to the operator already coded by category. To select a billing category, callers from touch-tone telephones could press digits in response to a menu of options. Callers from non-touch-tone telephones would have to identify their choices in an alternative manner, perhaps with speech which would be classified via an automatic speech recognition device. Most of the human factors issues for this system involved prompting each caller to say the right thing at the right time (without prior training), so that the automatic speech recognition device could accurately classify the utterance.[16]

A simulation was designed to establish how people behave when faced with a particular set of automatic speech recognition prompts.[17] Callers were told that they were participating in a study. Their attention was focused on the evaluation of an experimental AT&T service. Since it was important to learn how people respond to unexpected speech prompts, callers were not told about the prompts before they dialed the experimental service.

A prototype system that imitated the automatic operator service was constructed to allow investigation of caller reactions to recorded instructions. The system simulated a telephone network complete with dial tone and audible ringing, and provided a means of emulating the real network as necessary. It could concatenate and play a large number of prerecorded announcements, and the composite messages could be cued according the caller's response.

If the caller successfully dialed the service's number and delivered a spoken response to establish the billing category, the system produced a busy signal and played an announcement that asked the caller to try again later. This happened twice; the third time the caller succeeded in reaching the service. This repeated experience with the prompts for speech allowed observation of an individual caller's learning.

Among the central questions answered in this study were the following:

1. What is the best *way to cue* callers to talk (i.e., to speak their billing category)? Three different forms of

**Table IV. Responses to Wording**

| Initial prompt | Percent appropriate response (240 participants) |
|---|---|
| "Please say . . ." | 78 |
| "Please say the type of call you want to make . . ." | 53 |
| "Please request a type of call by saying . . ." | 76 |
| "This is your automatic operator; please say . . ." | 74 |

cue were tested: (a) an announcement asking the caller to speak a word to indicate a billing preference (the *initial prompt*) followed by an auditory menu of choices and punctuated by a prompt to speak the choice "now," (b) the initial prompt followed by an auditory menu and punctuated by an audible tone to signal the caller when to speak, and (c) an initial announcement instructing the caller to speak "at the tone," followed by the initial prompt, an auditory menu, and the audible tone. The percentages of appropriate responses (i.e., correct speech to indicate billing category) are shown in Table III. On the basis of a statistical analysis of these data, "at the tone" with a tone at the end of the initial prompt was selected for use in the rest of the study.

2. What is the best *wording* for the initial prompt that asks callers to say their billing choice? Four differently worded initial prompts were created that differed in level of explanation. The percentages of appropriate responses (i.e., correct speech to indicate billing category) are shown in Table IV. Statistical analysis revealed that one of the initial prompts, "Please say the type of call you want to make . . . ," was significantly worse than the others. "Please say . . ." was selected for use in the rest of the study because it was the shortest of the three prompts that elicited similar correct response rates.
3. Do *reprompts* for speech help when callers fail to respond appropriately? In the cueing method and initial prompt studies, if a caller did not say one of the menu items in isolation, a reprompt was given. Reprompts during the first trial (i.e., before the first busy signal) added 24 percent to the success rate. Reprompts during trials 2 and 3 added an additional 10 percent to the success rate.
4. Do callers show any effects of learning? In the cueing method and initial prompt studies, across all variables in this simulation, a dramatic learning effect was observed. While 58 percent of the callers responded appropriately on the first trial, 72 percent did so on the second trial and 74 percent did so on the third trial.

This simulation provided a unique opportunity to augment and simplify the subsequent live-traffic field trial. The simulation provided the pre-field-trial screening of announcements and reduced the candidate prompts by a factor of four. In addition, it provided two sources of information that field trials typically cannot: (1) observation of individuals learning from repeated experience and (2) in-depth caller reactions obtained via questionnaires as well as from behavioral data. Furthermore, the simulation allowed various protocols to be tested without interfering with actual call routing and without affecting callers' perceptions of current telephone service.

### Future Directions

In this article we have focused on the use of prototypes as user interface demonstration systems by human factors specialists for the design of high-quality user interfaces. Rapid prototyping is such an effective approach because it offers a realistic predevelopment view of a system in a form that encourages easy and frequent iteration. Such a system allows the human factors specialist to obtain behavioral data and opinions from users and to carry out usability testing. Our ability to develop the most realistic and testable prototype is directly related to the existence of a broad class of good rapid prototyping tools. As the tools that the human factors specialist has on hand increase in number and technical capability, so will our ability to fully exploit the prototypes. What types of innovations in rapid prototyping technology would allow the human factors specialist to create even more realistic and usable prototypes? We suggest the following:

- Prototyping tools should continue to be broadened to include the integration of multimode operation (voice, data, images, three-dimensional graphics, etc.) as well as the integration of alternative input devices (e.g., tablets, light pens, trackballs, automatic speech recognition, and features for the disabled such as special switches or puff/sip tubes). These enhancements will allow human factors specialists to develop testable user-interface prototypes that are more realistic given the rapidly advancing state of technology and the resulting advance in the sophistication of AT&T's products and services.
- Artificial intelligence (AI) technology is now practical and has already been built into (or used in) a number of products and services. The most typical commercial applications are expert systems. Since the uses of AI technology are vast (e.g., aid in configuring and repairing systems and intelligent on-line assistance and training), the continued integration of AI technology into prototyping tools must be encouraged.
- Although the simple linkage provided by current rapid prototyping tools is sufficient for creating prototypes that can be demonstrated, it is not always sufficient for creating prototypes realistic enough to be operated in a dynamic, unplanned way by users. A "programming by example"[18] capability would greatly enhance the dynamic and interactive character of prototypes, allowing for more valuable feedback from users as well as more extensive usability testing.



## Summary

The integration of prototyping into the product or service development process is an extremely valuable step, and is often a key to the success and overall quality of the product or service. In this article, the generality of the approach was illustrated by examples of graphical user interfaces for network management, telecommunications products such as cordless telephones, and an interactive speech interface to an operator service. Across these three examples, the major benefit gained through use of the prototypes was the same. They allowed the human factors specialists to address usability issues and evaluate alternative user-interface designs at an early stage and within a realistic context. Prototyping is most effective in improving the quality of a product or service when it is integrated as a *standard part of the design and development process*. Early use of prototypes to demonstrate features and determine users' preferences is a powerful method for developing high-quality products and services that meet users' needs.

## Acknowledgments

The authors wish to acknowledge the large contributions of Karen L. Hewitt (Human Factors Technologies Group, AT&T Consumer Products Laboratories), whose work on the HT series cordless telephone is described above, and Leon Gellman (User Interface and Quality Planning Department, AT&T Bell Laboratories), whose work on the automatic operator system is described above. We would also like to acknowledge the contributions of Diane Z. Lehder (Human Factors Technologies Group, AT&T Consumer Products Laboratories), who was responsible for some of the follow-up work described in the HT series cordless telephone example, and Leonard R. Kasday (User Interface and

Quality Planning Department, AT&T Bell Laboratories), who contributed heavily to the "Future Directions" section. In addition, we wish to thank those who reviewed this article and suggested improvements.

**References**

1. J. Connell and L. Brice, "Practical Quality Assurance," *Datamation*, August 15, 1984, pp. 93-100.
2. J. D. Gould and C. Lewis, "Designing for Usability—Key Principles and What Designers Think," *Communications of the ACM*, Vol. 28, No. 3, March 1985, pp. 93-100.
3. K. Lantz, *The Prototyping Methodology*, Prentice-Hall, Englewood Cliffs, New Jersey, 1986.
4. T. M. Gruenenfelder and W. B. Whitten II, "Augmenting Generic Research with Prototype Evaluation: Experience in Applying Generic Research to Specific Products," *Human-Computer Interaction—INTERACT '84*, B. Shackel, ed., Elsevier Science Publishers (North Holland), New York, pp. 913-917.
5. B. W. Boehm, *Software Engineering Economics*, Prentice-Hall, Englewood Cliffs, New Jersey, 1981.
6. J. S. Reitman Olson, W. B. Whitten II, and T. M. Gruenenfelder, "A General User Interface for Creating and Displaying Tree-Structures, Hierarchies, Decision Trees, and Nested Menus," *Human Factors and Interactive Computer Systems*, Y. Vassilou, ed., Ablex Publishing, Norwood, New Jersey, 1984.
7. J. S. Reitman, W. B. Whitten II, and T. M. Gruenenfelder, "Design and Evaluation of Computer Interfaces," *Tenth International Symposium on Human Factors in Telecommunications*, Helsinki, June 6-10, 1983.
8. S. P. Overmeyer, "Rapid Prototyping in the Design of Large-Scale Interactive Information Systems," *Human Factors Society Bulletin*, Vol. 30, No. 1, 1987, pp. 3-4.
9. K. J. Schmucker, "MacApp: an Application Framework," *Byte*, August 1986, pp. 189-193.
10. J. D. Naumann and A. M. Jenkins, "Prototyping: the New Paradigm for Systems Development," *MIS Quarterly*, September 1982, pp. 29-44.
11. W. A. S. Buxton and R. Sniderman, "Iteration in the Design of Human-Computer Interface," *Proceedings of the 13th Annual Meeting, Human Factors Association of Canada*, Point Ideal, Lake Abays, Ontario, September 4-6, 1980, pp. 72-81.
12. W. Swartout and R. Balzer, "The Inevitable Intertwining of Specification and Implementation," *Communications of the ACM*, Vol. 25, No. 7, July 1982, pp. 438-440.
13. R. E. A. Mason and T. T. Carey, "Prototyping Interactive Information Systems," *Communications of the ACM*, Vol. 26, No. 5, May 1983, pp. 347-354.
14. N. S. Anderson and J. R. Olson, eds., "Methods for Designing Software to Fit Human Needs and Capabilities," *Proceedings of the Workshop on Software Human Factors*, National Academy Press, Washington, D. C., 1985.
15. H. R. Hartson and D. Hix, "Human-Computer Interface Development: Concepts and Systems," *ACM Computing Surveys*, Vol. 21, No. 1, March 1989, pp. 5-92.
16. R. W. Bennett et al., "Speaking to, from, and through Computers: Speech Processing Technologies and User-Interface Design," *AT&T Technical Journal*, Vol. 68, No. 5, September/October 1989, pp. 16-30.
17. L. H. Gellman and W. B. Whitten, II, "Simulating an Automatic Operator Service to Optimize Customer Success," *Proceedings of the International Symposium on Human Factors in Telecommunication*, The Hague, The Netherlands, May 24-27, 1988.
18. B. A. Myers and W. Buxton, "Creating Highly-Interactive and Graphical User Interfaces by Demonstration," *SIGGRAPH '86 Conference Proceedings*, Special Interest Group on Computer Graphics, Dallas, August 18-22, 1986.

*(Manuscript received June 21, 1989)*