

# The evolution of user-centered focus in the human-computer interaction field

by J. Karat  
C. M. Karat

About 20 years have passed since the first conferences dedicated to human-computer interaction (HCI) were held. In that time many changes have occurred in how we think about making use of data gathered from users of technology to guide the process of designing and developing new hardware and software systems. Throughout this process there has been a productive dialog among academic and industry-based researchers and usability engineering practitioners. Academic research has provided insights into methods for understanding and modeling user behavior, and industry has provided a wide range of exciting technologies for consideration by researchers in HCI. This paper looks at the evolution of the field from the behavioral science perspective. We consider the evolution of the field within professional groups, such as the Association for Computing Machinery Special Interest Group on Computer-Human Interaction (ACM SIGCHI) and the International Federation for Information Processing Technical Committee (IFIP TC13), academic departments (primarily in computer science departments), and industry (primarily within IBM). In this paper we offer a view of this journey of 20 years, along with some visions and challenges of the future.

We offer a historical perspective on the development of the human-computer interaction (HCI) field over the last 20 years. We do not attempt to identify or detail the origins of the field, but instead focus on how it has changed during the time that we have con-

sidered ourselves specialists in this area. We feel that our timing was good, if not quite perfect. Although we were not fortunate enough to have stumbled on the pioneering work of Doug Engelbart or the exciting work going on at Xerox PARC (Palo Alto Research Center) during our graduate studies (see Pew<sup>1</sup> for a good overview), we were completing our graduate work in psychology at a time (1982) when technology companies were beginning to feel a need to better understand how people interact with computers. There were no degrees in HCI in those days—just people who had a strong interest in making technology more valuable for a wider audience and who had a belief that behavioral science could play an important role in advancing this cause.

We will not try to be comprehensive in this brief paper. In the 20 years since the first conference on human factors in computing systems was held in Gaithersburg, Maryland in 1982, which led to the first Association for Computing Machinery Special Interest Group on Computer-Human Interaction (ACM SIGCHI) Conference in 1983 and the first INTERACT Conference in 1984, thousands of research papers have been published on a wide range of topics. The field has thrived and struggled with its multidisciplinary nature. Research has focused on raw technology (such as pointing devices and speech recognition), on interface styles (such as command languages, menu designs, and direct manipulation), on

©Copyright 2003 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

the use of technology in the workplace and at home, and on methods for design of usable systems. HCI research has examined a variety of application domain areas such as health care, education, finance, home use, and entertainment. Additionally, the impact of technology developments such as pervasive computing and the Internet have led to an increased examination of human issues such as privacy, trust, and security. Our focus here is on how the role of the user of technology has changed and evolved in research and development within the computer industry. In 20 years, we have seen a movement from focusing on specialists (e.g., computer operators or programmers) to examining how technology impacts us all. The user of technology tended to be viewed as the human (perhaps error prone) necessary to complete some task with a system. Now we are attempting to view the user more complexly: as a human in a social system in which the computer plays an increasingly important role.

The history of HCI is one in which we have experienced a gradual shift from attempts by different parts of the community to focus on their own narrow view of the field to a more cooperative effort to understand what it means to build systems that people value. When the first author of this paper first came to IBM Research, there was a “theory” group working to extend behavioral science theory to describe human interaction with a particular type of technology (mostly desktop computers). The group focused on the kind of behavioral theories that were largely individual and cognitive; that is, they looked mostly at the computer as a tool for individual use and were primarily interested in how cognitive activities such as learning and problem-solving developed in computer users. There were also various technology groups focused on issues such as display resolution and speech recognition. For them, HCI was about developing technology that was better without worrying specifically about measuring empirically how valuable it might be to humans. For the behavioral scientists, how useful the behavioral theories would be for design was not a primary concern. It was simply, optimistically perhaps, assumed that theory would be valuable. For the technologist, the uses to which the new technology would be put was not a primary concern. It was assumed that better technology would find uses. We do not think that there is anything wrong with these positions, but, over the years, the HCI community has looked for ways in which dialog between behaviorally and technically oriented researchers might lead to more productive end results.

## Defining human-computer interaction

In 1982, a conference called “Human Factors in Computer Systems” was held in Gaithersburg, Maryland.<sup>2</sup> This conference has come to be regarded as the first major conference devoted to human performance issues in computer system design, development, and use. The event was followed by the renaming and refocusing of the Special Interest Group on Social and Behavioral Computing (SIGSOC) within ACM to SIGCHI<sup>3</sup> and by the ACM SIGCHI conferences that have been held since 1983. Similar developments within the international community led to the formation of a Technical Committee on Human-Computer Interaction within the International Federation for Information Processing (IFIP)<sup>4</sup> and to the IFIP TC13 INTERACT conferences that followed from 1984. At the time, a Computer Systems Technical Group already existed within the Human Factors and Ergonomics Society (and still does), but this group was and remains more narrowly focused both in its membership and in its focus on the evaluation of human-machine interfaces.<sup>5</sup> Although most persons doing evaluation of human-computer dialog were members of the Human Factors Society in 1982, most working on the design and development of new technologies were not. In our opinion, the SIGCHI and INTERACT conferences were successful (both immediately and over the past 20 years) because they provided a forum for synergy between those focused on the human side of HCI and those focused on the technology side.

We consider this multidisciplinary approach a key component of the field of human-computer interaction. Although membership in worldwide professional organizations related to HCI exceeds 10000 academicians, researchers, and practitioners, very few of these people actually have degrees in HCI. Lack of such degrees is related to the fact that there are still very few universities that offer such a degree, more out of the difficulty in clearly defining the components of HCI as a multidisciplinary field than from lack of interest. Backgrounds of researchers and practitioners include a range of behavioral sciences that includes psychology, anthropology, and sociology, along with computer science and other science and engineering disciplines. These different communities are brought together by the shared goal of producing technological systems that are better for humans, and by the shared belief that no one view of how to do so holds the answer.

Calling this desire a shared goal is perhaps a bit of an overstatement, and describing the HCI field as a happy community of different disciplines glosses over some fundamental difficulties in human communication. In the early days of the field, there were clear differences in the types of contributions that different communities thought were most important to the field. Behavioral scientists thought the field needed more data obtained from observations (mostly in the laboratory) of humans completing tasks using technology. Technologists were focused on technologies for making interactions more “natural,” such as direct manipulation or speech recognition. Behavioral scientists, generally trained in observation and evaluation, were either unwilling or uneasy in developing and iterating on the design of user interfaces or systems for humans to use. It took years before they realized that “owning the design” was important to being effective in development. Technologists saw the advances in computational power as central and were unwilling or uneasy with trusting mere users to evaluate their contributions.

A look at the papers from the first and from the most recent SIGCHI and INTERACT conferences provides an interesting view of the field in its early days (see Table 1). For the early conferences (CHI'83, INTERACT'84, and CHI'85), we noted four main classes of contributions: papers about systems intended to enhance some human activity (systems = 17 percent), papers about methods for the development of systems (methods = 15 percent), papers offering analysis of some activity employing technology (usage description = 26 percent), and papers comparing two or more design approaches for systems to accomplish some task (UI comparison = 14 percent). Additional topics included studies of command languages, menu design and usage, formal specifications and task analysis approaches, and various ergonomic topics, such as mouse button design or screen lighting.

We would like to point out a few things in the comparison of early and recent conferences. First, there is a significant increase in the number of papers focused on new or unique systems. Although these papers have always been a part of the field, they seem to make up an increasing part of the publications at the major conferences. (Note that the number of publications in the field has been relatively constant over the years.) Second, there has been a significant decrease in the number of papers about methods—from about 15 percent to about 6 percent. As we will discuss later in this paper, we view this decrease as

Table 1 Analysis of paper topic distribution in early and recent ACM SIGCHI and IFIP INTERACT conferences

Paper Topics	Percentage of Papers in CHI'83, INTERACT'84, and CHI'85 (n = 207)	Percentage of Papers in CHI'01, INTERACT 2001, and CHI'02 (n = 214)
System	17	35
Method	15	6
Usage Analysis	26	27
UI Comparison	14	20
Others	28	12

having to do with a maturation of the methods employed in the design of usable systems. This maturation results in an increase in the use of methods, but a decrease in the description of the methods in the scientific literature. We return to this theme throughout the paper.

### Developing a definition of usability

A key factor to maintaining cooperation in the multidisciplinary community was development of a clearer sense of the scope of overall goals. How could we define what it meant to develop technology that was better for humans? Initially, the maxim “easy to use,” which means requiring the minimum cognitive and physical effort, was the guiding objective for evaluation. Evaluation papers at the early conferences were largely about measuring and broadly comparing different interface components, such as menu designs or input techniques. Command language terms were studied,<sup>6,7</sup> breadth and depth of menu structures were compared,<sup>8</sup> and the new technologies of direct manipulation and what-you-see-is-what-you-get (WYSIWYG) text editors were explored.<sup>9,10</sup> There was some sense that performance measures on narrowly specified tasks were not the complete answer to understanding what made a better system, but it would take some time for practitioners in the field to develop an agreement on a broader definition that might include more qualitative measures and a more complete notion of technology in use.

In 1985, a committee of the Human Factors and Ergonomics Society was beginning to explore the development of standards and guidelines for human-computer interfaces. The scope of the project was to focus on guidelines for aspects of the interface that were primarily under software control. Thus we were concerned with issues of dialog structure, help

and training, and information presentation, but not with screen resolution or keyboard ergonomics. We were also interested in the design process itself and in what kind of guidance might be created for developing software systems. The general idea was to take the results of human factors studies of human-computer interaction and present them as guidance for the designers of software systems. The fact that work on this project (ANSI/HFES 200)<sup>11</sup> and on a closely related project within the International Organization for Standardization (ISO 9241)<sup>12</sup> is still underway after over 15 years of ongoing effort is testimony to how difficult it can be to provide guidance through standards on the development of usable systems and even to define exactly what we mean by usability.

The eventual output of these efforts will include software system design guidance at a variety of levels. For the purposes of this paper, we want to mention just two of the parts included in the ISO 9241 standard. These parts present a kind of high-level guidance that we do not commonly associate with the term “standard,” and cover principles for good dialog design (Part 10) and guidance on how to specify and evaluate usability in different contexts (Part 11). Part 10 is entitled “Dialogue Principles” and includes the following seven principles:

- Suitability for the task
- Self-descriptiveness
- Controllability
- Conformity with user expectations
- Error tolerance
- Suitability for individualization
- Suitability for learning

In the text of the standard, these principles are illustrated by examples rather than by strict definition. For example, we are told that a dialog is suitable for a task when it supports the users in effective and efficient completion of the task, but we are not told exactly what this suitability means. In general, the examples provided in the standard are qualitative illustrations of the principles (e.g., “the dialogue should avoid forcing unnecessary steps” as an application of the suitability principle), rather than prescriptions for design.

The information in Part 11 (entitled “Guidance on Usability”) complements Part 10. At the heart of Part 11, there is a definition for the term “usability,” and then there is discussion (again through example rather than by narrow prescription) of how to go

about measuring it. The definition of usability included in the standard is:

*Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use.*

The importance of these two documents lies not as much in the specifics of the guidelines included in the various parts as it does in the acknowledgment that usability is complex and context-dependent. Designing for usability should be seen as involving both attendance to high-level principles (such as those in Part 10) and selection of particular approaches that are determined by a context of use that is broader than we have generally attended to. The acceptability of any software product is not dependent solely on surface interface features, but also on the way a system fits within a use context. Whereas this might seem obvious to some, we believe that the community of system designers, including usability specialists, continues to struggle with difficult questions of how to turn this obvious fact into specific approaches for dealing with context of use in design. We may know that context is important, but we still do not know exactly what to do about it.

We mention this because the evolution of HCI as a field can be seen as related to an ongoing broadening of the scope of what designing for usability means in system development. In some ways this statement is really unfair because both traditional systems design and human factors approaches have not ignored context of use in design. “Know the users and their tasks” has always been the declared first step of any rational design process. However, the specific approaches developed so far have been much more tuned to well-specified narrow contexts than to the realities of the workplace or of home life. It is still fairly common for us to settle for some simplified notion of the context of use for a system obtained from some source outside of the development environment as capturing our understanding of users and tasks. For example, it is still common to rely on data gathered from questionnaires to provide a profile of intended users of a system. Although these data can be useful at a high level, they do not provide the sort of detailed understanding of the context of use necessary for design. For example, knowing the average age or education level of the users under-specifies the design space. It does not tell the designer enough to help in reaching decisions on design details.



As use of computing technology spread outside of the United States, the HCI field became influenced by traditions that studied work more broadly. Although the Scandinavian countries had long been influenced by worker participation in the design of work and workplace tools, the skills involved in this activity emerged more strongly around 1994. The terms “participatory design” and “user-centered design” first appear in the keyword indexes of CHI’91, and “ethnography” enters in CHI’94. Analysis of human activity has always been a major activity within the field (as can be seen in Table 1), but it then started to take a broader form. It was associated more with broad involvement of users in design and less with laboratory examination of subjects in narrow tasks.

Additionally, a new pair of HCI-related conferences emerged, namely the Computer Supported Cooperative Work (CSCW) and the European Computer Supported Cooperative Work (ECSCW) conferences. Briefly stated, the optimism of the 1980s concerning the amount of progress we could make in the HCI field by using strictly formal methods faded a bit. Models (e.g., GOMS, a model based on a theory of cognitive skills involved in human-computer interactions) were only successful for modeling small task components. Laboratory studies could not capture the richness of technology in the workplace. Such tools would be a part of HCI research and practice, but other approaches were necessary.

## User-Centered Design

Professionals whose work focus has been on the development of usable systems have changed what they call themselves over the past 20 years in rough correspondence with the scope and focus of their work. In the early 1980s, human factors specialists were thought to work on interface components (e.g., icon design, menu layout, terminology). Generally they did carefully controlled evaluations in laboratory settings. As both the specialists themselves and the organizations that employed them gained a greater appreciation for the broader context into which any piece of software fits, many specialists looked for (and continue to look for) different techniques to apply and different labels for what they did. Names commonly used to identify a usability specialist changed from human factors specialists to usability engineers to User-Centered Design (UCD) specialists as the focus of their activities moved from product assurance testing to integrated product design and development.

We do not claim that such name changes are a mere fad, even though they may reflect very little in the way of real content changes for the underlying activities involved in developing usable systems. Names can be valuable in communicating what one considers important. Within the general field of human-computer interaction, we have many attitudes and approaches (e.g., participatory design, contextual inquiry, UCD). Though these names are all loosely

---

**Professionals whose work focus  
has been on the  
development of usable  
systems have changed what  
they call themselves.**

---

united by a commitment to developing more usable human-computer systems, the focus of activity is reflected in the title. For participatory design, the activities focus on worker involvement in the design process. For UCD there is no general agreement in the field as to the focus areas; different organizations take different approaches to UCD. In general, the activities are focused on understanding the needs of the user as a way to inform design (see Vredenburg et al.<sup>13</sup>).

If the range of activities that one carries out changes as greatly over time as it has for usability specialists, it can be useful to change the label for the job holder. For example, while we were perfectly content with “human factors specialist” in our job titles in the mid-1980s, the associations of this title with the sorts of things human factors people do made it valuable to us to change the title to usability engineer as our organizations took a broader view of usability. Including the term “engineer” in the title was intended to suggest that it was possible to view the activity as driven by data and methodology. The data in this case were viewed as coming from users more than from handbooks. The methodology involved targeting behavioral objectives in iterative design. In this case, value in the name change came from a perceived role difference by those we had to work with in different research or development teams. We were “engineers” with data and tools to contribute to design. In the last few years, the computer-human interaction community has generally adopted the title of UCD for describing the attitudes and approaches used for developing usable systems. As mentioned

earlier, a wide range of approaches are included under this label, and it is clear that these approaches are much broader than the approaches to usability common in the 1980s.

We have also generated different terms to describe the systems we develop in correspondence to the scope of use we focus on during the development. Groupware systems did not suddenly appear; many software tools supported group work long before the term was invented within the field. However, identifying group work as a particular context and giving new labels to tools and communities that focus on this area does make sense. It seems to us that progress in usability of systems for particular contexts is much more likely when the features of that context are brought into focus than when they are left in the background or ignored. Similarly, to the extent that there are identifiable context elements that distinguish systems developed for entertainment or home use rather than for office work, we expect to see specialized names given to the activities associated with understanding the users and their goals in these environments.

In a world in which everything is a little different from anything else, when does it make sense to distinguish something from other (similar) things? For the design of usable work systems, we think the answer to this question is when the context of use suggests that user acceptance is sufficiently different to call for different approaches to design. It might be that systems of type X call for a usability approach of type Y. Such an approach has been sketched out for a range of application domains suggesting, for example, that there might be different approaches appropriate for domains as different as entertainment and health care,<sup>14,15</sup> but much work remains to be done in this area.

At the 1996 ACM SIGCHI Conference held in Vancouver (CHI'96), the first author of this paper organized and moderated a panel entitled "User-Centered Design: Quality or Quackery." What was meant in selecting this title? Should we not expect that the people attending a conference on human factors in computer systems be united in viewing user involvement in design as central to producing quality systems? Unfortunately, there are reasons to ask the question. Rather than becoming more clearly defined as the computer-human interaction community matured, we saw the term "user-centered design" as becoming akin to "family values" in nature—a concept to which everyone subscribed, but for which there

seemed to be no agreed-upon definition. As one panelist (Dennis Wixon) pointed out during the panel session, if we cannot define what UCD is, then we are faced with allowing virtually anything to be called a UCD process. Is UCD a term that describes anything that usability specialists do, or is it a specific set of

---

### **What best defines User-Centered Design?**

---

techniques drawn from a larger set of activities that may be a part of system design? Can (or must) we tolerate ambiguity in the definition, or is precise definition of the UCD process necessary?

In the time leading up to the conference, the first author did an informal survey of usability specialists in about 20 companies to determine just how clear everyone was in their thinking about UCD. We had reason to believe that there would be a diversity of opinions. For many years we had been involved with attempts to gather information about design activities that contributed to usable systems as a part of company, national, and international human-computer interaction standards efforts. We knew that even though some groups might be able to define processes that work for them, there is great difficulty in getting different groups to agree on a single process. For example, are the techniques of good design for a follow-on release of a word processor the same as those for an electronic medical record system? Maybe they are when sufficiently abstracted, but it is not clear that the resulting abstractions are of much use to the practitioner trying to decide what to do and how much resource to allocate to do it.

In the informal survey, we asked whether companies had defined a UCD process. In doing so we imagined that there would be a number of possible answers—that companies did have clearly defined processes that varied in scope and detail, or that there was no specific user involvement in the development process. In most cases the response to the survey was that attempts to define such a process were underway. Often, the process definitions were considered as confidential—companies saw some potential for competitive advantage in having defined the best process. Without attempting to catalog the details, we tried to discover the extent to which they included

high-level attitudes or principles (such as those in Part 10 and Part 11 of ISO 9241) or specific techniques or approaches (such as, “Focus groups with at least 10 users will be conducted before product requirements are finalized.”). When pressed for details, most of those we surveyed confessed that there was little secret in their processes, saying that the definition effort was mostly aimed at making the content of their UCD process more explicit and fitting this process into the development process of their organization.

Recently, Vredenburg et al.<sup>16</sup> conducted a more formal survey to look at UCD practice. In this survey they offered a definition of UCD as involving “active involvement of users for a clear understanding of user and task requirements, iterative design and evaluation, and a multidisciplinary approach.” Although they found that actual measures of the effectiveness of the overall UCD process were lacking, there does seem to be a steady maturing of the key notions and the methods for making UCD fit within system development. This work gives us considerable reason to be optimistic about UCD becoming a more clearly defined practice. It also helps to articulate important research issues, such as the need to better understand the complexities of system design in a pervasive computing world.

What best defines UCD? The seminal chapters by Gould<sup>17</sup> and by Whiteside et al.<sup>18</sup> in the *Handbook of Human-Computer Interaction* of 1988 provide excellent coverage of UCD although neither one used UCD as a label. Gould presents four high-level principles for good design. The principles include early and continual focus on users (suggesting direct user involvement in the design process), early and continual evaluation, iterative design and development, and integrated (whole system) design. The techniques (i.e., the generally required steps involved in reaching the goals specified in the principles) are illustrated by examples such as “go out into the field and learn about the work organization.” Whiteside et al. provide more specific guidance for a small (but very difficult) part of the overall development activity: setting measurable usability objectives and working explicitly to achieve them. If the things advocated in these chapters are performed, the likelihood of successful system design is raised. Are there obstacles? Sure—and plenty of them—from the technical (e.g., being able to deliver required information to users in a timely fashion) to the organizational (e.g., being able to create and maintain teams of peo-

ple with the range of expertise necessary to solve complex design problems).

How clearly is UCD defined? During the CHI’96 panel on UCD, Dennis Wixon offered a definition of UCD as a process that sets users or user data as the criteria by which a design is evaluated or as a generative source of design ideas. We find this definition to be a useful distinction to keep in mind. However, another of the panelists (Martin Rantzer) argued that it was too high a level for many organizations. Although high-level principles might be enough to guide some organizations, they were not sufficient to obtain buy-in from most management for funding actual development. Many organizations still require some sort of proof that it is necessary to talk to users in order to understand what to build. In taking exception to the word “centered” in UCD during the panel discussion, Michael Atwood (now at Drexel University) suggested that design involves participants, each of whom brings both knowledge of a domain and ignorance of other domains to the process; no one perspective is really at the center. We must admit that we sometimes become nervous about others taking user-centered design to mean that users (or usability people) should be in control of design rather than being equally critical participants. It results partly from the actual words used in the title—whether it is “user” or “performance” centered. Placing anything at the “center” can lead to a perception that the other elements are somehow less important. We think that the Vredenburg et al. study suggests that the field is becoming clearer about the methods, but that there is still a great deal of work to do before the practice could be considered as fully mature.

We believe that the HCI field generally agrees on the principles of UCD and that the field believes that these principles apply to the development of all types of systems. However, experience with the UCD survey along with experience in trying to describe the best practice in various standards activities suggests that we do not have broad agreement on the techniques for achieving the principles. Knowing what to do is still an important problem. Many blank stares can still be seen when “usability engineering” is suggested as an integrated part of a project. Development requires the coordinated action of a number of people, and coordinating that action means that an individual has to have some sense of what is going to be done when, and at what cost in resource (both effort and money). We do believe that we have come a long way from the old days in understanding

what human-factors practitioners can (and should) do to contribute to creating usable systems, but it is currently more in the heads of experienced practitioners than in the HCI literature.

### Current and emerging focus areas for HCI

What has happened in 20-plus years of HCI work? We see four main trends that we briefly characterize below.

The focus of HCI research and practice has moved from *interface* to *interaction*. This trend started fairly

---

**Design has emerged  
as a central focus  
within HCI.**

---

early in the history of the field, though there remains some confusion because there is still a tendency to at least talk about designing the interface rather than designing interaction (or, more currently, designing the user experience). Although there will always be work to do in designing the interface between human and tool, we have somewhat successfully moved toward an interface style that works and are moving more to broader questions of how technology fits within a broad range of human needs. These questions indicate how we interact with technology in our lives, more than they pertain to what the interface looks like.

The pace of change in computing systems has shifted from *slow* changes in technology to *rapid* changes. We began our studies in the early days of personal computers at a time when issues associated with the usability of mainframe systems were still relatively important. The technology would stand still long enough for us to study it and its users in great detail. The pace of technology change brings with it some new challenges. Although behavioral science provides us with many tools and theoretical frameworks for observing behavior, they are generally tuned for use in a fairly stable environment and not for providing design advice in a rapidly changing one. Newer approaches, such as integrating the use of scenarios into HCI design work,<sup>19</sup> are part of the shift to approaches that fit the technological reality.

The population impacted directly by the technology has moved from a few users to everyone. Relatively speaking, system operators and administrators are not the target audience they once were. We moved from the small group of users for whom operating a computer was their main task, to knowledge workers looking for productivity tools, to computers as providers of entertainment as well as information. We are as concerned about how the technology impacts the very young and the aging as much as we are interested in office workers. Associated with this concern is a shift to looking beyond individual activity and cognition for appropriate theoretical guidance to HCI work. Theoretical approaches such as activity theory<sup>20</sup> or distributed cognition<sup>21</sup> move the focus away from concentrating on the individual that has dominated HCI toward the roles of humans in a larger world.

There has been a broadening of our conception of the role of technology from focusing on office work *productivity* to considering a broad range of *use*. One indication of the rise in importance of looking beyond productivity alone has been the attention given to evaluating the total user experience within UCD.<sup>13</sup> This does not mean that office productivity does not deserve or receive attention; it is the focus of a great deal of human activity. However, there is an acknowledgment that activity and technology have value outside of the office and outside of narrow views of work. Value-sensitive design<sup>22</sup> and considering the application of UCD methods to entertainment applications<sup>15</sup> have become increasingly important.

For the authors, UCD defines iterative processes whose goal is the development of usable systems. There is general agreement that this goal is achieved through involvement of potential users of a system in system design. In this we feel that we must be somewhat less specific about what role users play in the process than some argue for. For example, in the participatory design community, approaches have been developed to enable the users to take active roles in many design activities. In the context in which these techniques were developed (Scandinavian countries with strong labor unions), users have the right to design their work environments. It is likely that techniques derived from this experience might need to be modified to fit use contexts that are different. System design is ultimately a partnership between developer and user, and the level of partnership between user and developer is a factor that will vary. For example, the design of games is something that should involve the participation of potential



users, but UCD experts would not approach game design in the same task-oriented way as they would approach an application design in banking.

*Design* has emerged as a central focus within HCI. The HCI field is increasingly interested in the design process: how it should proceed and who needs to take part in it. The field has discovered that there are many views on what the term “design” means and different ways to view the measurement of quality of a design. For most people within the HCI community, “good design” still implies a sense of “fitting a purpose” that can be empirically validated. As the field moves more toward considering systems that people value for purposes other than as tools, we are finding an increasing need to consider aesthetics and other factors that can contribute to the value of a system or an artifact. The contribution of “satisfaction” to the usability equation is receiving increasing attention.

## Concluding remarks

We suggest that UCD is a good label under which to continue to gather knowledge of how to develop usable systems. It captures a commitment that the usability community supports—that users must be involved in system design—while leaving how this is accomplished fairly open. We think that there is knowledge and skill associated with practicing UCD, but we do not think the field has yet captured the necessary training in academic programs (see Karat and Dayton<sup>23</sup> for more discussion of this). All techniques that involve users in design can be called UCD. The challenge is to build up an understanding of how and when each is appropriate. The HCI field must keep in mind that developing usable software involves more than involving users, and the usability community must keep in mind that many perspectives need to be balanced in fielding a new system. Developing new systems or applications is engineering and is always done within a context of design trade-offs and limited resources. We are certainly justified in focusing some attention on user involvement, but we must do so while keeping in mind the difficult necessity of multidisciplinary communication in design.

## Cited references

1. R. W. Pew, “The Evolution of Human-Computer Interaction: From Memex to Bluetooth and Beyond,” in *The Human-Computer Interaction Handbook*, J. A. Jacko and A. Sears, Editors, Lawrence Erlbaum Associates, Mahwah, NJ (2002), pp. 1–15.
2. *Human Factors in Computer Systems*, Institute for Computer Sciences and Technology, U.S. National Bureau of Standards (now NIST), Washington, DC (1982).
3. Special Interest Group on Computer-Human Interaction (SIGCHI), Association for Computing Machinery, at <http://www.acm.org/sigchi/>.
4. Technical Committee on Human-Computer Interaction, International Federation for Information Processing (IFIP), at [http://www.csd.uu.se/ifip\\_tc13/](http://www.csd.uu.se/ifip_tc13/).
5. Computer Systems Technical Group, Human Factors and Ergonomics Society, at <http://cstg.hfes.org/index.html>.
6. J. Rosenberg, “A Featural Approach to Command Names,” *Human Factors in Computing Systems—CHI’83 Conference Proceedings*, A. Janda, Editor, ACM, New York (1983), pp. 116–119.
7. R. Kraut, S. Hanson, and J. Farber, “Command Use and Interface Design,” *Human Factors in Computing Systems—CHI’83 Conference Proceedings*, A. Janda, Editor, ACM, New York (1983), pp. 120–123.
8. R. Teitelbaum and R. Granda, “The Effects of Positional Constancy on Searching Menus for Information,” *Human Factors in Computing Systems—CHI’83 Conference Proceedings*, A. Janda, Editor, ACM, New York (1983), pp. 150–153.
9. A. Lee and F. Lochovsky, “Enhancing the Usability of an Office Information System through Direct Manipulation,” *Human Factors in Computing Systems—CHI’83 Conference Proceedings*, A. Janda, Editor, ACM, New York (1983), pp. 130–134.
10. J. Ziegler, H. Hoppe, and K. Fahnrich, “Learning and Transfer for Text and Graphics Editing with a Direct Manipulation Interface,” *Human Factors in Computing Systems—CHI’83 Conference Proceedings*, M. Mantei and P. Oberton, Editors, ACM, New York (1986), pp. 72–77.
11. ANSI/HFES 200 Standard, *Ergonomic Requirements for Software User Interfaces*, an extension of ISO 9241, developed in the Human Factors and Ergonomics Society and final document approved by the American National Standards Institute. See P. Reed and P. Billingsley, “Software Ergonomics Comes of Age: The ANSI/HFES 200 Standard,” *Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting* (1996), pp. 323–327.
12. ISO 9241 Standard, *Ergonomic Requirements for Office Work with Video Display Terminals*, International Organization for Standardization, Geneva (1998).
13. K. Vredenburg, S. Isensee, and C. Righi, *User-Centered Design: An Integrated Approach*, Prentice-Hall, Upper Saddle River, NJ (2002).
14. J. Karat, “User-Centered Software Evaluation Methodologies,” in M. Helander, T. K. Landauer, and P. Prabhu, Editors, *Handbook of Human-Computer Interaction*, Elsevier Science Publishers, Amsterdam (1997), pp. 689–704.
15. J. Karat, “Beyond Task Completion: Evaluation of Affective Components of Use,” in *The Human-Computer Interaction Handbook*, J. Jacko and A. Sears, Editors, Lawrence Erlbaum Associates, Mahwah, NJ (2002), pp. 1152–1164.
16. K. Vredenburg, J. Mao, P. Smith, and T. Carey, “A Survey of User-Centered Practice,” *Human Factors in Computing Systems—CHI’2002 Conference Proceedings*, ACM, New York (2002), pp. 471–478.
17. J. D. Gould, “How to Design Usable Systems,” in *Handbook of Human-Computer Interaction*, M. Helander, Editor, Elsevier Science Publishers, Amsterdam (1988), pp. 757–789.
18. J. Whiteside, J. Bennett, and K. Holtzblatt, “Usability Engineering: Our Experience and Evolution,” in *Handbook of*

- Human-Computer Interaction*, M. Helander, Editor, Elsevier Science Publishers, Amsterdam (1988), pp. 791–817.
19. J. Carroll, *Making Use: Scenario-Based Design of Human-Computer Interactions*, MIT Press, Cambridge, MA (2000).
  20. B. Nardi, *Context and Consciousness: Activity Theory and Human-Computer Interaction*, MIT Press, Cambridge, MA (1996).
  21. P. Dourish, *Where the Action Is: The Foundations of Embodied Interaction*, MIT Press, Cambridge, MA (2001).
  22. B. Friedman, P. Kahn, and J. Hagman, "Hardware Companions?—What Online AIBO Discussion Forums Reveal about the Human-Robotic Relationship," *Human Factors in Computing Systems—CHI'2003 Conference Proceedings*, ACM, New York (2003), pp. 273–280.
  23. J. Karat and T. Dayton, "Practical Education for Improving Software Usability," *Human Factors in Computing Systems—CHI'95 Conference Proceedings*, R. Mack, I. Katz, and L. Marks, Editors, ACM, New York (1995), pp. 162–169.

*Accepted for publication June 18, 2003.*

**John Karat** *IBM Research Division, Thomas J. Watson Research Center, 19 Skyline Drive, Hawthorne, New York 10532 (jkarat@us.ibm.com).* Dr. Karat is a research staff member at the Watson Research Center. He received his B.S. degree in mechanical engineering from Lehigh University, and a Ph.D. in cognitive psychology from the University of Colorado at Boulder. He conducts human-computer interaction (HCI) research in a variety of areas, including privacy, personalization, and information management. Dr. Karat is chairman of the International Federation for Information Processing Technical Committee on HCI (IFIP TC13), North American editor for *Behaviour and Information Technology*, and Editor-in-Chief of the Kluwer Academic Publishers series on HCI. He is a member of the ACM SIGCHI Executive Committee and has been actively involved in ACM CHI and DIS, and the IFIP INTERACT conferences in the HCI field.

**Clare-Marie Karat** *IBM Research Division, Thomas J. Watson Research Center, 19 Skyline Drive, Hawthorne, New York 10532 (ckarat@us.ibm.com).* Dr. Karat is a research staff member at the Watson Research Center. She received her B.A. degree in psychology with honors from Stanford University and a Ph.D. in social psychology from the University of Colorado at Boulder. She conducts human-computer interaction (HCI) research in the areas of privacy, personalization, and conversational interface technologies. She is an international expert in cost-justifying human factors work and develops innovative user interface designs and methodologies. Dr. Karat has published numerous articles in professional and technical journals and has authored several chapters in recently published books. As an editorial board member of the ACM magazine *interactions* and the *Interacting with Computers Journal* of the British Computer Society, and as a technical program committee member of the CHI, HFES, and INTERACT conferences, she provides leadership and maintains an active network of communication with HCI professionals in the field.