User Interfaces for All: New Perspectives into Human–Computer Interaction

Constantine Stephanidis

This chapter introduces the notion of User Interfaces for All, elaborates on the motivating rationale and examines its key implications on Human–Computer Interaction. The underlying vision of User Interfaces for All is to provide an approach for the development of computational environments catering for the broadest possible range of human abilities, skills, requirements and preferences. Consequently, User Interfaces for All should not be conceived as an effort to advance a single solution for everybody. Instead, it is a new perspective into Human–Computer Interaction, seeking to unfold and reveal challenges and insights, and to instrument appropriate solutions for alleviating the current obstacles to the access and use of advanced information technologies by the widest possible end-user population.

INTRODUCTION

Human–computer interaction (HCI) is concerned with the design, implementation, and evaluation of interactive computer-based systems, as well as with the multidisciplinary study of various issues affecting this interaction. The aim of HCI is to ensure the safety, utility, effectiveness, efficiency, accessibility, and usability of such systems. The user interface is the part of an interactive system, application, or telematic service with which the user comes into contact cognitively, perceptually, and physically.

In recent years, HCI has attracted considerable attention by the academic and research communities, as well as by the information technology and telecommunications (IT&T) industry. An early argument in favor of HCI studies, from the perspective of system development, was the fact that the programming effort for the user interface, in many cases, exceeds 50% of the total programming effort required for the development of the entire system. Though this is generally true, today there are other more compelling reasons why HCI is becoming critical in the emerging information society.

The proliferation of computer-based systems and applications, and the widespread use of emerging telematic services, have introduced new dimensions to the issue of HCI. Human activities are increasingly becoming mediated by computers, thus replacing the initial view of computers as scientific devices with the percep-
tion of computers as tools for productivity enhancement. With the advent of the information society, the creation of new virtual spaces is likely to bring about yet another paradigm shift, whereby computers are no longer conceived as mere business tools, but as integrated environments, accessible by anyone, anytime, and anywhere. Therefore, in the information era, it is important to develop high-quality user interfaces, accessible and usable by a diverse user population with different abilities, skills, requirements, and preferences, in a variety of contexts of use, and through a variety of different technologies.

This chapter introduces the concept of user interfaces for all as a new perspective into HCI that aims at anticipating, analyzing, and tackling the challenges posed in the development of interactive applications and telematic services in the emerging information society. The chapter analyzes the ongoing paradigm shift, its driving factors and trends, and the challenging requirements it poses. The historical context of user interfaces for all is briefly described, and the implications on HCI are unfolded. Furthermore, the chapter provides an account of the new perspectives and insights that this concept brings about in the design, implementation, and evaluation of interactive software accessible and usable by all users in a variety of contexts of use.

THE PARADIGM SHIFT

The radical innovation in the IT&T sectors, the ever-growing demand for information access, and the diffusion of computers across industry sectors and application domains, are the driving forces of the ongoing paradigm shift toward an information society.

The Trends

From the early calculation-intensive tools that were prevalent in the 1970s, computer-based systems are progressively becoming tools for communication, collaboration, and social interaction among groups of people. From a specialist's device, the computer is being transformed into an information appliance for the citizen in the information society. It follows, therefore, that designers increasingly have to provide information artifacts to be used by diverse user groups, including people with different cultural, educational, training, and employment background, novice and experienced computer users, the very young and the elderly, and people with different types of disabilities (see chaps. 2 and 3, this volume).

Similarly, the context of use is changing. The "traditional" use of computers (i.e., scientific use by the specialist, business use for productivity enhancement) is increasingly being complemented by residential and nomadic use, thus penetrating a wider range of human activities in a broader variety of environments, such as the school, the home, the marketplace, and other civil and social contexts. As a result, information artifacts should embody the capability to interact with the user in all those
contexts, independently of location, target machine, or run-time environment. Usability in such "nontraditional" usage contexts is likely to prove a harder target to meet than in the case of the workplace (Stephanidis & Akoumianakis, 1996).

Finally, technological proliferation contributes with an increased range of systems or devices to facilitate access to the community-wide pool of information resources (see, e.g., chap. 5, this volume). These devices include computers, standard telephones, cellular telephones with built-in displays, television sets, information kiosks, special information appliances, and various other "network-attachable" devices. Depending on the context of use, users may employ any of the aforementioned to review or browse, manipulate, and configure information artifacts, at any time.

**HCI Challenges**

The trends outlined previously bring about radical implications on the current and future HCI research focus and perspectives. Until recently, interaction in the context of interactive systems was considered as a dialogue between humans and computer-based embodiments of real-world analogies, conveyed predominantly in a visual manifestation. Design was largely conceived as the art of constructing this dialogue at different levels (i.e., semantic, syntactic, or lexical). Design knowledge, therefore, reflected the accumulated experience in performing design activities at various levels using suitable tools and techniques.

The aforementioned conceptions have changed significantly due to radical technological progress brought about to fulfil the increasing demand for information by an ever-growing information "client base." As a result, the scope and range of computer-mediated human activities have been expanded to facilitate more sophisticated business tasks of the "knowledge" worker, as well as a broad range of residential activities by nonexperts. Additionally, the emergence of novel application domains (e.g., World Wide Web) and advanced interaction technologies (e.g., mobile devices, wearable equipment) progressively shifts the focus of design to consider virtualities beyond the traditional desktop computer, and a variety of, until now, unconventional usage contexts. Consequently, user interactions with computer-embodied artifacts have become more complex to envision and design.

It is frequently argued that the hardest target to achieve en route to the information society is the design of new computer-embodied artifacts to facilitate the broad range of emerging activities (see chaps. 29 and 30, this volume; see also Winograd, 1997). For this to be attained, HCI should revisit some of the basic assumptions that have shaped recent work and progress. Some of these are briefly discussed next.

**The "Average" Typical User.** One of the basic guidelines for HCI design is to study the user. Many recent methodological frames of reference introduce this guideline as a milestone for subsequent stages in the process. Additionally, there has been a wide variety of instruments for capturing requirements and studying us-
ers, such as questionnaires (e.g., *Usability Context Analysis*; Bevan & Macleod, 1994), checklists (e.g., *EVADIS*; Reiterer & Oppermann, 1995), formal and semiformal methods (e.g., *cognitive walkthrough*; Lewis & Polson, 1990; Lewis, Polson, Wharton, & Riesman, 1990), as well as various *ethnographically inspired approaches* (Simonsen & Kensing, 1997).

Notwithstanding the relative merits of each method or tool, it is important to mention that their usefulness should be carefully judged. This is not due to the validity of the techniques, but rather to the changing conceptions and contexts in studying users. More specifically, in the context of the emerging distributed and communication-intensive information society, users are not only the computer-literate, skilled, and able-bodied workers driven by performance-oriented motives, nor do users constitute a homogeneous mass of information-seeking actors with standard abilities, similar interests, and common preferences with regard to information access and use. Instead, our conception of users should accommodate all potential citizens, including the young and the elderly, residential users, as well as those with situational or permanent disability. Consequently, it becomes increasingly complex for designers to know the users of their products, and compelling to design for the broadest possible end-user population. This raises implications on design methodology and instruments, as well as on the technical and user-perceived qualities to be delivered.

*The Business Environment.* Another assumption that has influenced HCI throughout its short history has been the context of use in which computer-based designs are encountered. Due to the unlimited business demand for information processing, the HCI community has progressively acquired a bias and a habitual tendency toward outcomes (i.e., theories, methods, and tools) that satisfy the business requirements and demonstrate performance improvements and productivity gains. However, since the early 1990s, analysts have been concerned with the increasing residential demand for information, which is now anticipated to be much higher than its business counterpart. Consequently, designers should progressively adapt their thinking to facilitate a shift from designing tools for productivity improvement, to designing computer-mediated environments of use. This leads to the next point, which addresses the need to extend the prevailing metaphors for interaction design to suit the changing requirements.

*The Desktop Computer Embodiment.* The desktop embodiment of current interfaces is perhaps the most prominent innovation delivered by the user interface software industry, including the tool development sector. However, the diffusion of the Internet as an information highway and the proliferation of advanced interaction technologies (e.g., mobile devices, network attachable equipment) signify that many of the tasks to be performed by humans in the information age will no longer be bound to the visual desktop. New metaphors are likely to prevail in the design of the emerging virtual spaces, reflecting the broader type and range of computer-mediated human activities. Arguably, these metaphors should
encapsulate an inherently social and communication-oriented character in order to provide the guiding principles and underlying theories for designing more natural and intuitive computer embodiments. Consequently, the challenge lies in finding powerful themes and design patterns to shape the construction of novel communication spaces. At the same time, it is more than likely that no single design perspective, analogy, or metaphor will be adequate for all potential users or computer-mediated human activities. Design will increasingly entail the articulation of diverse concepts, deeper knowledge, and more powerful representations to describe the broader range and scope of interaction patterns and phenomena.

USER INTERFACES FOR ALL—AN OVERVIEW

The term user interfaces for all denotes an effort to unfold and reveal the aforementioned challenges, as well as to provide insights and instrument appropriate solutions in the HCI field. The underlying vision of user interfaces for all is to offer an approach for developing computational environments that cater to the broadest possible range of human abilities, skills, requirements, and preferences. Consequently, user interfaces for all should not be conceived as an effort to advance a single solution for everybody, but rather, as a new perspective on HCI that alleviates the obstacles pertaining to universal access in the information society.

The roots of user interfaces for all can be traced in the notions of universal access and design for all. The term design for all (or universal design—the terms are used interchangeably) is not new. It is well known in several engineering disciplines, such as, for example, civil engineering and architecture, with many applications in interior design, building and road construction, and so on (Story, 1998). Though existing knowledge may be considered sufficient to address the accessibility of physical spaces, this is not the case with information society technologies, where universal design is still posing a major challenge. Universal access to computer-based applications and services implies more than direct access or access through add-on (assistive) technologies, because it emphasizes the principle that accessibility should be a design concern, as opposed to an afterthought. To this end, it is important that the needs of the broadest possible end-user population are taken into account in the early design phases of new products and services.

Several previous efforts addressed the accessibility of computer-based applications and services by disabled and elderly people through the a posteriori adaptation of interactive software. This amounts to a “reactive” approach, whereby assistive technology experts attempted to respond to contemporary technological developments by building accessibility features into interactive applications, as a result of specific user requirements. Such efforts to account for accessibility are, however, mainly governed by intuition and usually follow ad hoc procedures that may lead to suboptimal solutions with respect to user requirements (see chap. 6, this volume).

Although the reactive approach to accessibility may be the only viable solution in certain cases (Vanderheiden, 1998), it suffers from some serious shortcomings,
especially when considering the radically changing technological environment, and, in particular, the emerging information society technologies. First, reactive approaches, based on a posteriori adaptations, though important to partially solve some of the accessibility problems of people with disabilities, are not viable in sectors of the industry characterized by rapid technological change. By the time a particular access problem has been addressed, technology has advanced to a point where the same or a similar problem re-occurs. The typical example that illustrates this state of affairs is the case of blind people’s access to computers. Each generation of technology (e.g., DOS environment, windowing systems, and multimedia) caused a new “generation” of accessibility problems to blind users, addressed through dedicated techniques, such as text-to-speech translation for the DOS environment, off-screen models, and filtering for the windowing systems.

In some cases, adaptations may not be possible without loss of functionality. For example, in the early versions of windowing systems, it was impossible for the programmer to obtain access to certain window functions, such as window management. In subsequent versions, this shortcoming was addressed by the vendors of such products allowing certain adaptations on interaction objects on the screen.

Finally, adaptations are programming-intensive, which raises several considerations for the resulting products. Many of them bare a cost implication that amounts to the fact that adaptations are difficult to implement and maintain. Minor changes in product configuration, or the user interface, may result in substantial resources being invested to rebuild the accessibility features. The situation is further complicated by the lack of tools to facilitate “edit-evaluate-modify” development cycles (Stephanidis, Savidis, & Akoumianakis, 1995).

Due to the aforementioned shortcomings of the reactive approach, there have been proposals and claims for proactive strategies, resulting in generic solutions to the problem of access. One such proposal is user interfaces for all, which constitutes an attempt to apply, exemplify, and specify the principles of universal access and design for all in the context of HCI.

Proactive strategies entail a purposeful effort to build access features into a product, as early as possible (e.g., from its conception, to design and release). In the context of HCI, user interfaces for all advocates such a proactive paradigm for the development of systems accommodating the broadest possible end-user population (Stephanidis, 1995). In other words, user interfaces for all seeks to minimize the need for a posteriori adaptations and deliver products that can be adapted for use by the widest possible end-user population (adaptable user interfaces). This implies the provision of alternative interface manifestations depending on the abilities, requirements, and preferences of the target user groups.

Design for all is often criticized on the grounds of practicality and cost justification. In particular, there is a line of argumentation raising the concern that “many ideas that are supposed to be good for everybody aren’t good for anybody” (Lewis & Rieman, 1994, Section 2.1, Paragraph 3). However, universal design in IT&T products should not be conceived as an effort to advance a single solution for everybody, but as a user-centered approach to providing products that can automati-
ally address the possible range of human abilities, skills, requirements, and preferences. Another common argument is that universal design is too costly (in the short term) for the benefits it offers. Though the field lacks substantial data and comparative assessments as to the costs of designing for the broadest possible population, it has been argued that (in the medium to long term) the cost of inaccessible systems is comparatively much higher, and is likely to increase even more, given the current statistics classifying the demand for accessible products (Bergman & Johnson, 1995; Vanderheiden, 1990). What is really needed is economic feasibility in the long run, leading to versatility and economic efficiency (see chap. 28, this volume; see also Vernardakis, Stephanidis, & Akoumianakis, 1997).

The concept of user interfaces for all was originally introduced in 1995 (Stephanidis, 1995), following the results of several research initiatives in the context of collaborative project work cofunded by the European Commission. The ACCESS project (1994–1996; see Acknowledgments) was one of the first efforts to develop user interface development tools that would provide a vehicle toward user interfaces for all (ACCESS Technical Annex, 1993). ACCESS delivered a powerful methodology, the unified user interface development methodology, and a novel platform for designing and implementing user- and use-adapted interactions (see Part IV, this volume). The unified user interface methodology was demonstrated in specific applications in the field of disability in the context of the ACCESS project. Subsequently, it was applied by the AVANTI project (1995–1998; see Acknowledgments), which developed a unified Web browser inherently accessible by different categories of users with disabilities (see also chap. 25, this volume). Besides these demonstrations, the concepts addressed in the ACCESS project (e.g., development of nonvisual interaction metaphors, encapsulation of alternative dialogue patterns, platform abstraction, generation of user interface implementation through executable specifications rather than programming) proved to have wider significance. In particular, many of these concepts have been endorsed and adopted by recent accessibility initiatives, industrial organizations and consortia, and nonmarket institutions. In subsequent chapters of the volume, the reader will find the details about some of these developments.

IMPLICATIONS ON HCI

Universal access, and by implication user interfaces for all, introduce challenges that cover the broad spectrum of HCI work, including design, development, and evaluation. Some of these challenges are briefly addressed in an attempt to reveal new research issues that should guide ongoing and future efforts in this direction.

User interfaces for All and HCI Design

In the relatively short history of HCI, several approaches have been developed to address the design of computer-based interactive artifacts. The paradigms of human factors evaluation (Salvendy, 1997), cognitive science (see chap. 8, this volume), and more recently, user-centered design (Norman & Draper, 1986) have
been the most prominent. Human factors evaluation and cognitive science have been criticized with respect to the underlying scientific ground and the means (e.g., instruments, methods, and tools) they use for achieving the set objectives. The HCI literature reports many ongoing debates (e.g., Bødker, 1991; Carroll, 1991; Nardi, 1996; Suchman, 1987; Winograd & Flores, 1987), and offers an insight into the diversity that characterizes the field.

The realization of the shortcomings underpinning the previous two design approaches led many researchers to recognize that, on the one hand, traditional large-scale user testing is both costly and suboptimal, whereas, on the other hand, any attempt for a generalized model or theory for designing interactions between humans and machines is simply not feasible (see chap. 8, this volume). Instead, techniques are needed that focus on the requirements of end-users, and provide early feedback to design, so as to reduce the cost of design defects and to meet specific usability objectives. The term coined to characterize this approach was user-centered design, and the first comprehensive collection of articles on the topic appeared in Norman and Draper (1986). The normative perspective of user-centered design is to fulfill the need for usability by providing techniques that foster tight design-evaluation feedback loops, iterative prototyping, early design input, and end-user feedback. In subsequent years, and due to the compelling need to cost-justify usability throughout a product’s life cycle, the field moved toward a variety of techniques, generally referred to as inspection-based evaluation (Nielsen, 1993), that, though inexpensive, are less formal in their conduct and deliverables. Following several success stories in the use and cost justification of these techniques, the consolidated experience gave rise to a generally applicable process model for constructing human-centered systems. Human-centered design is documented in a draft International Standards Organisation (ISO) document (ISO 13407, 1997), which provides a principled approach and guidelines for attaining usability.

In addition to the developments leading to user-centered design, there have been several proposals for remedying the shortcomings of human factors evaluation and information-processing psychology. These proposals stem from the social sciences, and aim to improve the experimental grounds for HCI design by emphasizing the need for analytical, prescriptive frameworks for studying users in specific contexts of use. Existing attempts toward the objective just discussed make suggestions for:

- Better exploitation of the existing knowledge and science base by utilizing the experimental ground of developmental approaches to psychology (e.g., cultural/historical/work psychology) and the social sciences (e.g., anthropology, sociology, humanities; see chap. 8, this volume).
- Broadening the range, or even extending the scope of information-processing psychology with concepts from developmental approaches to HCI, such as activity theory (Bødker, 1989, 1991; see also chap. 10, this volume), language/action theory (Winograd, 1988), situated action models (Suchman, 1987), and distributed cognition (Norman, 1991).
The normative perspective adopted in these efforts is that interactions between humans and information artifacts should be studied in specific *social contexts* and take account of the distinctive properties that characterize them. Despite this common commitment to the study of context, the aforementioned alternatives differ with regard to at least three dimensions, namely the unit of analysis in studying context, the categories offered to support a description of context, and the extent to which each approach treats actions as structured prior or during human activities (Nardi, 1996).

In the context of user interfaces for all, user-centered design, as well as the previously mentioned emerging approaches, have several contributions to make. First of all, the human-centered protocols and tight design-evaluation feedback loop of user-centered design provide a new insight into how interactive systems can be developed. Such an insight aims to replace the technocentric practices of the current paradigm with a human focus, which will help and guide designers to identify and attain accessibility, usability, and other *quality of use* targets (see chap. 18, this volume). To this effect, however, user-centered design needs to advance the current collection of tools so as to provide the means to study context and to complement existing artifact-oriented practices with analytical and process-oriented instruments.

Developmental approaches to HCI design, rooted in cultural psychology, the social sciences, and the humanities, hold the promise to provide contributions in this direction by offering richer tools for analytical design and prescriptive frameworks for studying varieties of context of use.

**User Interfaces for All and User Interface Development**

The second major implication of user interfaces for all is on user interface development. This entails several questions related to the sufficiency of both the currently available architectural models for guiding the construction of user interface software, as well as the overall engineering paradigm for user interface development.

**Traditional Architectural Models for User Interface Software.** A challenging issue relates to a suitable reference model for user interface architectures facilitating design for all in HCI. Early work, following the appearance of graphical user interfaces, focused on window managers, event mechanisms, notification-based architectures, and toolkits of interaction objects. Such architectural models were quickly supported by mainstream tools, becoming directly encapsulated in the prevailing user interface software and technology. Today, all available user interface development tools support object hierarchies, event mechanisms, and callbacks as the basic implementation model.

In addition to these, there have been other proposals for architectural models of interactive software, which, however, have not gained the level of acceptance originally anticipated. The focus of these proposals has been to extract a reference model from concrete user interface architectures, in order to classify existing prototypes
and to guide the construction of user interface software. Specifically, the Seeheim metamodel (ten Hagen, 1991) and its successor, the Arch metamodel (UIMS Developers Workshop, 1992), were proposed to foster the principle of separation between interactive code (i.e., dialogue control) and noninteractive code (i.e., functional core) of computer-based applications. These models influenced early research efforts in the area of user interface management systems (UIMSs) (Myers, 1995). There have also been alternative, implementation-oriented proposals, some of which have resulted from mainstream tools. For instance, the model view controller (MVC) (Goldberg, 1984) originates from the architectural abstractions underpinning the Smalltalk-80 programming model, whereas it has also influenced the Java’s pluggable look-and-feel architecture. The PAC model (Coutaz, 1990) is similar to MVC, though not yet supported by any commercial tool.

These models fall short of addressing several of the requirements for user interfaces for all. First of all, they do not explicitly account for the notion of accessibility of an interactive application. Thus, they do not address issues such as multiple platform environments, toolkit integration, platform abstraction, and so on, which arise from the proliferation of novel interaction platforms and diversity in usage patterns. Second, these models offer no account of user interface adaptation (Dieterich, Malinowski, Kühme, & Schneider-Hufschmidt, 1993), which is a central theme in design for all in HCI. As a consequence, key decisions such as what and when to adapt are not addressed, whereas the components that are needed to drive adaptations at the level of the user interface are totally missing. Third, existing architectural models offer implementation-oriented views of user interface architectures; this limits the role of design by not addressing how design knowledge can be propagated to the development and implementation phases. As a result, the application of these models in current HCI practices leads to reimplementations (reactive approach) rather than instantiation of an alternative design (proactive approach).

**Development Techniques.** Having presented the implications on the user interface software architecture, the next item addressed is whether or not prevailing development practices can facilitate the development of user interfaces for all. To assess this, we review some of the currently popular techniques for user interface development and consider their underlying focus. Some of these techniques have already been supported in commercially available user interface tools, others are embodied in tools that are currently available as public domain software, whereas yet another cluster is still in prototype versions. In considering these techniques, our objective is to assess their suitability for user interfaces for all. This means that we do not seek to compare them against any particular implementation approach but, instead, we wish to consider whether or not, and to what extent, the specific techniques considered make provisions for the fundamental requirements of user interfaces for all.

Currently, there are various interface development techniques and tools (Myers, 1995). For the purposes of our assessment, these are classified under six distinctive categories, namely presentation based, physical task based, demonstra-
tion based, model based, abstract objects and components, and declarative fourth-generation languages (4GLs). Presentation-based techniques include graphical construction tools, such as Visual Basic™ and TAE plus™. Physical task-based techniques lead to a specific sequence of user actions and include TAG (Reisner, 1981) and UAN (Harton, Siochi, & Hix, 1990). Demonstration-based techniques are similar to graphical construction methods, with the exception that they allow the interactive definition of a physical interface instance through an example, or a demonstration. Demonstration-based techniques have been embedded in systems such as Peridot (Myers, 1988), Pavlov (Wolber, 1996), and DemoII (Fischer, Busse, & Wolber, 1992). Model-based techniques represent a more recent effort based on the notion of generating interactive behaviors from suitable models about users, tasks, platforms, and the environment. A comprehensive retrospective account of model-based technology can be found in (Szekely, 1996). Examples of model-based tools include HUMANOID (Szekely, Luo, & Neches, 1992) and MASTERMIND (Szekely, Sukaviriya, Castells, Muthukumarasamy, & Salcher, 1995). Abstract objects/components are techniques that support alternative physical realizations through either object abstractions, such as meta-widgets (Blattner, Glinert, Jorge, & Ormsby, 1992), or component-ware technologies, such as Active X™ by Microsoft and JavaBeans™ by SunSoft. Finally, declarative 4GL methods are typically found in some UIMSs, such as SERPENT (Bass, Hardy, Little, & Seacord, 1990) and HOMER (Savidis & Stephanidis, 1998).

Because a key ingredient of user interfaces for all is the capability to encapsulate alternative interactive behaviors, it follows that development methods closer to the physical level of interaction (e.g., presentation based, physical task based and demonstration based) are less suited. On the other hand, techniques that focus on higher level dialogue properties, and offer mechanisms for articulating alternative interactive components, stand a better chance and could be considered as candidates for integrating unified user interface development facilities.

User Interfaces for All and Evaluation Cycles

In addition to the design and development implications identified earlier, user interfaces for all pose several challenges in terms of evaluation. Evaluation in this context entails both the formative element through which design feedback is provided to early prototypes, and the summative account of high-fidelity prototypes. The related challenge seems to have a dual substance: On the one hand, it addresses the instruments for evaluation, whereas on the other, it raises considerations on the process and the overall iterative software development cycle.

At the level of the instrument, we need to identify what is to be evaluated or measured before we can decide how to do it. Clearly, user interfaces for all expand the targets of evaluation beyond performance-oriented accounts of usability. But these new targets remain to be defined. In addition, user interfaces for all introduce a com-

1TAE Plus™: http://www.cen.com/tae/
pelling need to address the evaluation of adaptation-related behavior of user interfaces, which, in turn, can only be effectively accounted for when studied in context. The effectiveness of existing evaluation instruments in order to capture context and facilitate such an insight remains an open issue (see chap. 17, this volume).

At the level of the design process, user interfaces for all pose an equally important challenge. User-centered design and other peripheral or similar approaches foster a collection of guidelines for maintaining a multidisciplinary and user-involving perspective into systems development. However, these models do not specify how designers can cope with radically different user groups whose requirements are not known a priori. Most important, they do not specify how to design for a universal community. In particular, with the advent of the Internet and the emergence of a highly distributed and collaborative computing paradigm, it is difficult for designers to anticipate who the user may be. For example, Web sites today can in principle be accessed by anyone possessing an Internet connection and a modem, irrespective of gender, age, educational background, and level of expertise. However, it is unlikely that this can be practically obtained with currently available applications. Furthermore, prevailing process models have not much to offer to systems developers in terms of practical means and recommendations toward such an objective.

It follows that the evaluation challenge posed by user interfaces for all is equally important as the design and development challenges. Whether or not HCI, in its current form and focus, can provide the required tools to address this challenge remains a question. It is believed, however, that before such a target is met, the community should face the fundamental question of what constitutes representative quality attributes in the emerging information age, and how these can be practically managed throughout the system development. The view that distinguishes between functional and nonfunctional qualities seems to be a useful starting point.

CONCLUSIONS

In this chapter, user interfaces for all is introduced as a new perspective into HCI, emphasizing some of the implications on the design, implementation, and evaluation of interactive software accessible to and usable by all users in a variety of contexts of use. The scope of user interfaces for all is broad and complex, because it involves issues pertaining to context-oriented design, diverse user requirements, as well as adaptable and adaptive interactive behaviors. This complexity arises from the numerous dimensions that are involved and the multiplicity of contributions in each dimension. Accordingly, the fields of science that might provide useful insights toward user interfaces for all are many and diverse. This is not different from the recent history of HCI, in the course of which the field has grown to develop strong links with several disciplines such as, for example, psychology (e.g., information processing and developmental psychology), linguistics, and the social sciences (e.g., anthropology, sociology, etc.). With regard to technology, user
interfaces for all links with recent advances in user interface software in terms of a variety of aspects including software engineering, advanced interaction platforms, tools for development, and so on.

This volume provides rich insights into the current state of the art, as well as into what remains a challenge for additional theoretical and applied research work.

REFERENCES


1. USER INTERFACES FOR ALL


