Beyond "Couch Potatoes": From Consumers to Designers

Gerhard Fischer
Center for LifeLong Learning & Design (L^3D)
Department of Computer Science and Institute of Cognitive Science
University of Colorado, Boulder

Abstract

The fundamental challenge for human-computer interaction (HCI) is to invent and design a culture in which humans can express themselves and engage in personally meaningful activities. Cultures are substantially defined by their media and tools for thinking, working, learning, and collaborating. New media change (1) the structure and contents of our interests, (2) the nature of our cognitive and physical tools, and (3) the social environment in which thoughts originate and evolve, and mindsets develop.

Unfortunately, a large number of new media are designed from a perspective of seeing and treating humans primarily as consumers. The possibility for humans to be and to act as designers (in cases in which they desire to do so) should be accessible not only to a small group of "high-tech scribes," but rather to all interested individuals and groups.

1. Introduction

Cultures are substantially defined by their media and their tools for thinking, working, learning, and collaborating. A large number of the new media are designed to see humans only as consumers. Television is the most obvious medium that promotes this mindset and behavior [44] and contributes to the degeneration of humans into "couch potatoes," for whom a remote control is the most important instrument of their cognitive activities. (A "couch potato" is a colloquial expression for a person who spends a lot of time on a couch consuming food and information in a passive fashion and who does not often engage in intellectual or physical activities).

Unfortunately, a consumer mindset does not remain limited to television, but in many cases is a model dominating our culture. In our educational institutions learners are often treated as consumers, creating a mindset of consumerism for the rest of their lives. Citizens often feel left out in the decisions by policy makers, denying them opportunities to take an active role. Computational media have the unique potential to let people be designers or assist them to incrementally become designers. Unfortunately, most current computational environments do not allow users to act as contributors and designers [41].

In this paper I will: (1) differentiate between consumer and designer perspectives; (2) discuss media support or limitations for these roles; (3) envision a future for HCI from this perspective; (4) illustrate some of our own work to address these issues; and (5) provide some evidence for the ubiquity of this framework for our society.

2. Images of Humans

2.1 A Consumer Perspective.

The Director of Research for Time Warner Entertainment, in his closing plenary address at CHI '95, articulated the design of a remote control to browse and efficiently select 500 or more TV channels as the basic challenge for the CHI community. Without a doubt, solving this problem is of great commercial interest to industries that regard humans as the ultimate consumers—but is it, or should it be, a focal issue for HCI?

In the early days of computing, humans were considered the "servants" of computers. As computers become cheaper, the basic economic criteria started to change and considerations of how to use computational power to augment and empower human beings were pioneered by some early visionaries [16,31]. These new ideas were neither known nor embraced by the community at large. The *Artificial Intelligence (AI)* community developed expert systems (such as MYCIN [5]), which were built as backward-chaining inference mechanisms. Although these systems could be built with reasonable effort, they were behaviorally unacceptable computational environments because they restricted knowledgeable and skilled human professionals such as doctors to answering yes or no to questions generated by the system.

Other disciplines such as *human factors* often considered humans as system components with specific characteristics such as limited attention span, faulty memory, and easy distractibility along with other undesirable characteristics. Early research in *human-computer interaction (HCI)* focused on interaction (idiot-proof systems, novices, naive users, and how walk-up-and-use systems could support their needs. Little consideration in the first decade of HCI research was given to the following perspectives:

- how we can support skilled domain workers who are neither novices nor naive users, but who are interested in their work and who see the computer as a means rather than as an end [27,39];
- how we can create co-evolutionary environments, in which users change, because they learn, and in which systems change, because users become co-developers and engage in end-user modification and programming [35];
- what it would mean to create intrinsically motivating computational environments [12], in which humans of all backgrounds would feel in control, get into the systems easily because of their low threshold but could grow in their capabilities over time because the systems also have a high ceiling [6].

2.2 A Designer Perspective.

The HCI community slowly started to understand that (at least) some human beings wanted to be more than consumers, and were not content with being novices or naive users indefinitely. New design philosophies introduced important new research objectives such as user-centered design [42], learner-centered design [10], and human-centered design [23]. Some members of the AI community started to consider the true goal of AI not as the replacement of human beings, but as the empowerment and augmentation of humans [2,51,52].

My arguments in this paper rest on the fundamental belief that humans (not all of them, not at all times, not in all contexts) want to be and act as designers (the concept "*designers*" in the context of this paper is used very broadly for a person who wants to act as an active participant and contributor in personally meaningful activities). Illich [30] (sharing this premise) has articulated the need for convivial tools and systems, which he characterized as follows: "convivial tools allow users to invest the world with *their* meaning, to enrich the environment with the fruits of *their* vision and to use them for the accomplishment of a purpose *they have chosen*" (emphasis added). Convivial systems encourage users to be actively engaged in generating creative extensions to the artifacts given to them and have the potential to break down the strict counterproductive barriers between consumers and designers.

2.3 Beyond Binary Choices.

By arguing for the desirability for humans to be designers, I want to state explicitly that there is nothing wrong with being a consumer and that we can learn and enjoy many things in a consumer role (e.g., listening to a lecture, watching a tennis match, attending a concert, seeing things in a movie on our couch that we are unable to see first hand). It is also a mistake to assume that being a consumer or being a designer would be a binary choice: it is rather a continuum ranging from passive consumer, to active consumer, to end-user, to user, to power users, to domain designer, to medium designer, all the way to meta-designer. Problems occur when (a) someone wants to be a designer but is forced to be a consumer, and (b) when being a consumer becomes a universal habit and mindset dominating a human life completely. For example: in thinking about the future (including the role of new media and new technologies for our future societies), the consumer asks: "Is a new future coming?" whereas the designer asks: "How can we invent and create a new future?" The designer understands that the future is not out there to be "discovered"—but it has to be *invented and designed*. Accepting this will raise the issue: *who* will design the future? I claim that the HCI community should not be content with either (1) restricting its efforts to the user interface or the computer part of HCI, or (2) reflecting and evaluating designs developed by other communities (e.g., the groups who give us 500 TV channels or artifacts over which we have no control). The HCI research community should not confine itself to a consumer role in the process of shaping our future knowledge society [13], in which they focus solely on some technical issues in the context of a world defined by others.

3. Technology and Media Support for Consumer and Designer Roles

Conviviality is a dimension that sets computers apart from other communication and information technologies (e.g., television) that are passive and cannot conform to the users' own tastes and tasks. Passive technologies offer some selective power (e.g., possibilities provided in "interactive" TV), but they cannot be extended in ways that the designer of those systems did not directly foresee.

Unfortunately, the potential for conviviality exists in many current computer systems only in principle. Many users perceive computer systems as unfriendly, uncooperative, and their use as too time consuming; they spend more time fighting the computer than solving their problems. Many users depend on specialists ("high-tech scribes") for help, and despite the fact that they deal with "soft"ware, they do not experience software as "soft" (i.e., the behavior of a system cannot be changed without reprogramming it substantially). The world of computing is separated into a population of elite scribes who can act as designers and a much larger population of intellectually disenfranchised computerphobes who are *forced* into a consumer role.

One of the biggest challenges for the HCI community will be to understand the fundamental difference between printed and computational media (and not being trapped by the "rear-view mirror" view [36] nor the "gift-wrapping approach" of new technologies [19]).

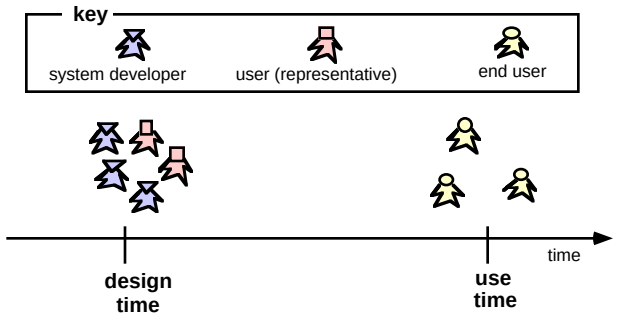

Figure 1: Design and use time

Figure 1 differentiates between two stages in the design and use of an artifact. At design time, system developers create environments and tools including help systems, guided tours, forms, etc., and they have to make decisions for users (who may want to be consumers or designers), for situational contexts and for task that they can only anticipate. For print media, a fixed context is decided at design time whereas for computational media, the behavior of a system at use time can take advantage of contextual factors (such as the background knowledge of a user, the specific goals and objectives of a user, the work context, etc.) *only known at use time*. The fundamental difference is that computational media have interpretive power: they can analyze and critique the artifacts created by users [22]—and users acting as designers will create artifacts off all kinds. The challenge is to build new innovative systems that allow the users to articulate contextual factors (e.g., in using a specification component [37] and/or infer this information from the environment), which will serve as objects for interpretation.

4. The Future of HCI from a Consumer/Designer Perspective

4.1 HCI Is More than User Interfaces.

Human-computer interaction is more than "screen-deep" [11,17,34]. The interface is important—but if we change only interfaces and not the systems behind them we will only be able to scratch the surface. HCI should strive for "interfaceless systems" supporting *human-problem domain interaction* [20] rather than human-computer interaction in which nothing stands between users and their tasks. It should be concerned with tasks, with shared understanding, with explanations, justifications, and argumentation about actions, and not just with interfaces. Although the usual concerns of interface designers (creating more legible types; designing better scroll bars; integrating color, sound, and voice; developing models of keystroke use [7]) are all important, they are secondary considerations. The essential challenges are improving the way people can use computers to work, think, communicate, learn, critique, explain, argue, debate, observe, decide, calculate, simulate, and design. The emphasis in the future has to be on humans and their tasks, not on computers.

4.2 Supporting Skilled Domain Workers.

Whereas computer scientists find computers intrinsically interesting and programmers like computers because they get to program, skilled domain workers regard computers as useful, sometimes even indispensable machines capable of helping them work on *their* problems more productively, creatively, and with greater pleasure [33,39]. As argued before, skilled domain workers are not novices or naive users, particularly with respect to domain concepts. They are people who have computational needs, who want and need to be designers, and who want to make serious use of computers in their work, but they are not interested in becoming professional programmers.

They are skilled and knowledgeable in their respective domains; they use computers by choice and over extended periods of time. To understand their use of computers requires a new orientation for many current HCI efforts. Rather than focusing on short-term, tool-oriented events, educational, social, and organizational needs (dealing with activities that range for days, months, and years) have to be investigated [8]. Traditional HCI efforts focus on evaluating the usability of the tools, rather than their applicability to the problem domain.

4.3 High-Functionality Applications: Creating Usable *and* Useful Systems.

Computer systems today are increasingly used to model parts of the world and not just to implement algorithms—and *the reality that they model is not user-friendly*. Systems that try to capture and model reality will therefore be complex, high-functionality applications. A general-purpose approach to creating tools aims to make them useful for a large number of problems. Systems with such a rich functionality are useful by offering a broad functionality, but they create serious usability problems for software designers and domain workers. Designers using these systems can no longer be "experts" (meaning individuals who know everything about a system) with respect to all existing tools. High-functionality applications create a "tool-mastery" burden that can outweigh the advantage of the broad functionality offered. The challenge is to change the "versus" relationship between usable and useful into an "and" relationship [17].

4.4 Closed versus Open Systems.

Closed systems (in which the essential functionality is anticipated and designed at design time; see Figure 1) are inadequate to cope with the tacit nature of knowledge and the situatedness of real-world problems. High-functionality applications offer too much functionality in the abstract, and they often fall short providing the

functionality and the knowledge needed for a particular situation. In our research we have carefully analyzed why simulation environments such as SimCity will not be used for real planning and working environments. SimCity acknowledges that users might want to design their own structures by providing the SimCity Urban Renewal Kit (an add-on module to increase user control), but it does not allow users to define new behavior. For example: if users notice that the crime rate is too high, they can build more police stations to fight crime, but they cannot increase social services or improve education to prevent crime. These empirical studies have led us to the following claims:

- *Software systems must evolve; they cannot be completely designed prior to use*. Design is a process that intertwines problem solving and problem framing [49]. Software users and designers will not fully determine a system's desired functionality until that system is put to use [27]. Process models that describe the different phases of the software life cycle need to take advantage of this fact [21].
- *Software systems must evolve at the hands of the users*. End users experience a system's deficiencies; subsequently, they have to play an important role in driving its evolution. Software systems need to contain mechanisms that allow end-user modification of system functionality [46].
- *Software systems must be designed for evolution*. Through our previous research in software design, we have discovered that systems need to be designed *a priori* for evolution [25] Software architectures need to be developed for software that is designed to evolve [15]. This is not only true for software, but for other artifacts as well, such as buildings [3]. Brand argues: "Almost no buildings adapt well. They're designed not to adapt; also budgeted and financed not to, constructed not to, administered not to, maintained not to, regulated and taxed not to, even remodeled not to. But all buildings (except monuments) adapt anyway, however poorly, because the usages in and around them are changing constantly".

5. Brief Summary of our Work

In our research in the Center for LifeLong Learning & Design [32], we combine the analysis and assessment of new computational media with the exploration of alternatives. Our work focuses specifically on (1) designing media and tools that users can learn, use, modify, and extend according to their needs; (2) supporting "human-problem domain communication" in which the computer disappears as medium in the background and at the same time tasks come to the foreground; and (3) providing opportunities for self-directed, lifelong learning.

5.1 Domain-Oriented Design Environments (DODEs).

DODEs (a research topic that we have pursued over the last decade [18,38]) can be built from appropriate substrates and support the design and construction of artifacts. They integrate working, learning, and collaborating among professionals from specific domains through mechanisms such as critiquing, catalogs of existing designs, argumentation, and end-user modifiability. Design environments not only can be used to help novice designers, they also are able to support designers as lifelong learners. DODEs are built upon substrates that support the creation of complex, open, and evolvable systems. Agentsheets [46] is an example of an existing substrate that can be used to develop educational interactive simulations. Agentsheets' visual programming approach allows a wide range of users to create these simulations in the context of self-directed learning, rather than being confined to a consumer role of dealing with what already exists. Within the context of DODEs, we have explored a number of specific effort to support a designer perspective.

Critiquing. Using design environments, designers create artifacts serving as externalizations of their thoughts. These artifacts can be critiqued by computational critics [22], increasing the "back-talk" of the design situation [50]. Only designers, not consumers, can engage in learning by doing, which has emerged as one of the most effective learning strategy. Critiquing (even in the simple form of a spelling corrector) requires computational media, because the artifacts produced by domain designers (users of the domain-oriented design environments) at use time (see Figure 1) need to be analyzed.

Organizational Learning and Organizational Memories. Organizational learning focuses on recording knowledge gained through experience (in the short term), and actively making that knowledge available to others when it is relevant to their particular task (in the long term). A central component of organizational learning is a repository for storing knowledge in an organizational memory. Individual projects serve organizational memory by adding new knowledge that is produced in the course of doing design work, such as artifacts, design rationale, and critiquing knowledge. Organizational memory is sustained in a useful condition through a combination of computational processes providing information and people using it, learning from it, and actively contributing to it (see Figure 2).

Dynasites [43] is a substrate for dynamic and evolvable web-based information spaces. Dynasites investigates computational support for collaborative working, learning, and knowledge construction by postulating that

information spaces supporting these activities should grow and be shaped over time by the people who use them. Dynasites information spaces are websites (they can be viewed through a web browser), but they differ from most websites because they are dynamic and evolvable by users. Dynasites information spaces are dynamic because their pages are built at "use" time (from a database) whereas typical web sites are static—their links and displays are determined at design time (see Figure 1).

Visual Agentalk and Behavior Exchange. While Dynasites is a substrate for creating textual, evolvable web-based information spaces, the Agentsheets *Behavior Exchange* [47] extends these evolvable spaces to include not only textual artifacts but computational artifacts including simulations and agents. Agents in Agentsheets are programmed in the programming language *Visual Agentalk* suitable for end-users, reducing the demands on end-users to migrate from consumers to designers.

5.2 The Seeding, Evolutionary Growth, Reseeding Model: A Process Model for Evolvable Systems.

We live in a world characterized by evolution—that is, by ongoing processes of development, formation, and growth in both natural and human-created systems. Biology tells us that complex, natural systems did not come into existence at once but must instead evolve over time. We are becoming increasingly aware that evolutionary processes are ubiquitous and critical for social, educational, and technological innovations as well.

The driving forces behind the evolution of these systems is their use by communities of practice in real-world problem solving as well as a changing world, specifically changes in technologies. The *seeding, evolutionary growth, and reseeding (SER) model* [21] is a process description of how this happens. The main aspects of the SER model can be characterized briefly as follows:

- A *seed* will be created through a participatory design process between environment developers and domain designers. It will evolve in response to its use in new design projects because requirements fluctuate, change is ubiquitous, and design knowledge is tacit.
- *Evolutionary growth* takes place as domain designers use the seeded environment to undertake specific projects for clients. During these design efforts, new requirements may surface, new components may come into existence, and additional design knowledge not contained in the seed may be articulated. During the evolutionary growth phase, the environment developers are not present, thus making end-user modification a necessity rather than a luxury.
- *Reseeding*, a deliberate effort of revision and coordination of information and functionality, brings the environment developers back in to collaborate with domain designers to organize, formalize, and generalize knowledge added during the evolutionary growth phases. Organizational concerns [26] play a crucial role in this phase. For example, decisions have to be made as to which of the extensions created in the context of specific design projects should be incorporated in future versions of the generic design environment. Drastic and large-scale evolutionary changes occur during the reseeding phase.

5.3 Assessment of the SER Model.

The SER model is motivated by how large software systems, such as Emacs, Unix, and Microsoft-Word, have evolved over time. In such systems, users develop new techniques and extend the functionality of the system to solve problems that were not anticipated by the system's authors (following the observation that any artifact should be useful in the expected way, but a truly great artifact lends itself to uses the original designers never expected). New releases of the system often incorporate ideas and code produced by users.

The development of the Linux operation system (as a prominent example of an *open-source software system)* [45] provides an interesting existence proof that reliable, useful, and complex systems can be built in a decentralized "Bazaar style" by many [48] rather than in a centralized, "Cathedral style" by a few. The Linux development model treats users as co-developers and is currently being tested in a number of new areas, such as: (1) *Netscape Communicator* [40]; (2) *Gamelan* [24], the first community repositories of Java-related information; and (3) *Educational Object Economy* [14] a collection of mostly completed Java applets designed specifically for education.

DODEs pose a major additional challenge to make the SER model feasible and workable: whereas the people in the above-mentioned development environments are computationally sophisticated and experienced users, DODEs need to be extended by domain designers (end-users with respect to computational media) who are neither interested in nor trained in the (low-level) details of computational environments. Domain designers are more interested in their design task at hand than in maintaining and evolving knowledge repositories per se. At the same time, important knowledge is produced during daily design activities that should be captured.

Figure 2 characterizes the duality and the distributed nature of knowledge: a specific user can learn (specifically learn in context and on demand) from a computational environment (which contains knowledge and tools contributed by many members of the community of practice), but if this user considers her/himself a designer, she/he will also contribute to the environment (assuming

mechanisms are available that allow her/him to do so with a reasonable effort). This perspective illustrates the concepts and need for co-adaptive systems: (1) users learn from the systems, (2) users acting as innovators, co-developers and designers adapt and evolve the systems, and (3) support for organizational learning allows users to share these adaptations with others.

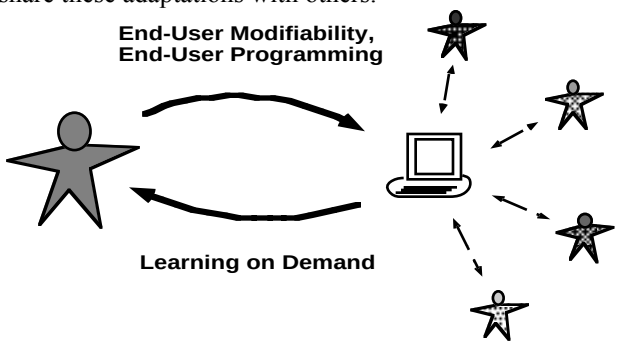

Figure 2: Duality between learning and contributing

6. The Ubiquity of the Consumer/ Designer Perspective

6.1 Learning and Education.

Even though the consumer/designer distinction provides many important challenges for HCI research and development of the future, it is a much more fundamental aspect of human behavior. To provide evidence for this claim, I briefly discuss its application to learning and education.

One of the most impoverished paradigms of education (based on the assumption that learners are consumers of knowledge) is a setting where a single, all-knowing teacher tells or shows presumably unknowing learners something they presumably know nothing about [4]. This model of education (which is widely practiced in our educational institutions) has led critics such as Illich [29] to the claims that our schools and universities are the "reproductive organs of a consumer society" and "people who are hooked on teaching are conditioned to be customers for everything else." As an alternative (and in analogy to Figure 2), we should reconceive classrooms and work environments as places in which subcommunities of mutual learners, acting simultaneously as learners and as designers and active contributors, in which peer-to-peer learning is supported and the teacher acts as a "guide on the side" rather than as a "sage on the stage" and in which courses are considered as seeds rather than finished products [19]. Such models of learning and education will require innovative computational environments that will share many of the requirements and challenges articulated in this article. The argument that consumerism is a mindset implies that there is no evidence that a "big switch" theory will succeed, meaning that a student who was educated as a passive consumer will suddenly switch to an active contributor. If the world of working and living requires a designer perspective (by relying on collaboration, creativity, definition and framing of problems, dealing with uncertainty, change, and distributed cognition), then our schools and universities need to prepare learners to be designers rather than consumers to have meaningful and productive lives in such a world.

6.2 Concerned Citizens—Taking Control of our Lives.

Citizens are being asked by initiatives in many parts of the United States and other countries to participate in community efforts (requiring substantial personal involvement) in the construction of sustainable futures. The Envisionment and Discovery Collaboratory [1] is an environment that combines physical game boards with computational simulations in order to support group decision-making processes, mutual learning processes, and the creation of shared understanding. The incremental design and construction of an externalized world provides opportunities for all involved stakeholders to act as active contributors and designers.

6.3 Motivation and Rewards.

Computational support mechanisms are necessary prerequisites, but not sufficient conditions to motivate people to become part of a "design culture". People must be motivated and rewarded for investing time and effort to become knowledgeable enough to act as designers. These rewards may range from feeling in control (i.e., independent from "high-tech scribes"), being able to solve or contribute to the solution of a problem, a passion to master a tool in greater depth, a ego-satisfying contribution to a group, and/or good citizenship to a community.

6.4 Cross-Cultural Perspectives—a Word of Caution.

I have written this paper coming from a specific cultural background: an embedding in European / American culture in which individualism may be more valued than collectivism. I believe that there are other cultural backgrounds in which people feel more comfortable than in my own being guided by a saga (i.e., by listening to a teacher, by accepting the opinion articulated or the artifact created by an "expert"). There is growing evidence that many themes of HCI (as it takes more cultural themes into account) will require more than simple translations efforts when ideas and systems are moved across cultural boundaries (themes explored by the "Culture, Communication and Creativity" Cognitive Science Laboratory at the Nara Advanced Institute of

Science and Technology [9]. I believe that we all become more aware of our own cultures as we encounter, reflect and discuss different cultures—an effort that I would consider an important element of the "designer culture" articulated in this paper.

7. Conclusions

HCI research and development have made very important contributions over the last decade. The HCI community has acquired a broad understanding of creating computational artifacts fitting better human capabilities and needs by creating theories and innovative systems [28]. To make the next step forward, the HCI community should accept the challenge to rethink computational media in broader contexts. My claim is that computational media can have the same fundamental impact on our individual lives and our societies as reading and writing had to move us from oral to literal societies. The true contribution of computational media might be to allow all of us to take on or incrementally grown into—in areas which we consider personally meaningful and important so we do not mind the additional effort—a designer role.

8. Acknowledgments

The author would like to thank the members of the Center for LifeLong Learning & Design at the University of Colorado, who have made major contributions to the conceptual framework and systems described in this paper. Hal Eden, Kumiyo Nakakoji, Frieder Nake, Rogerio dePaula, and Yunwen Ye have helped me with their feedback to improve this paper. The research was supported by (1) the National Science Foundation, Grants REC-9631396 and IRI-9711951; (2) NYNEX Science and Technology Center, White Plains; (3) Software Research Associates, Tokyo, Japan; (4) PFU, Tokyo, Japan; and (5) Daimler-Benz Research, Ulm, Germany.

9. References

1 Arias, E. G.; Fischer, G.; Eden, H., *Enhancing Communication, Facilitating Shared Understanding, and Creating Better Artifacts by Integrating Physical and Computational Media for Design,* In Proceedings of Proceedings of Designing Interactive Systems (DIS '97), Amsterdam, The Netherlands, pp. 1-12.

2 Bobrow, D. G., *Dimensions of Interaction*, In *AI Magazine* 1991, *12*, pp. 64-80.

3 Brand, S., *How Buildings Learn—What Happens After They're Built*; Penguin Books: New York, 1995.

4 Bruner, J., *The Culture of Education*; Harvard University Press: Cambridge, MA, 1996.

5 Buchanan, B. G.; Shortliffe, E. H., *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*; Addison-Wesley Publishing Company: Reading, MA, 1984.

6 Burton, R. R.; Brown, J. S.; Fischer, G., *Analysis of Skiing as a Success Model of Instruction: Manipulating the Learning Environment to Enhance Skill Acquisition* In *Everyday Cognition: Its Development in Social Context*; B. Rogoff and J. Lave, Ed.; Harvard University Press: Cambridge, MA - London, 1984; pp. 139-150.

7 Card, S. K.; Moran, T. P.; Newell, A., *The Psychology of Human-Computer Interaction*; Lawrence Erlbaum Associates, Inc.: Hillsdale, NJ, 1983.

8 Carroll, J. M., *Creating a Design Science of Human-Computer Interaction* In *Interacting with Computers*; Butterworth-Heinemann Ltd: , 1993; Vol. 5; pp. 3-12.

9 Cognitive Science Laboratory, *NAIST Cognitive Science Laboratory Home Page (English Version)*, At http://hawk.aist-nara.ac.jp/CCC/home-en.html, 1998.

10 Communications of the ACM, *Special Issue on 'Learner-Centered Design'*, In *Communications of the ACM* 1996, *39*,

11 Computer Science and Technology Board, *More Than Screen Deep: Toward Every-Citizen Interfaces to the Nation's Information Infrastructure*; National Academy Press: Washington, D.C., 1997.

12 Csikszentmihalyi, M., *Flow: The Psychology of Optimal Experience*; HarperCollins Publishers: New York, 1990.

13 Drucker, P. F., *The Age of Social Transformation*, In *The Atlantic Monthly* 1994, pp. 53-80.

14 East-West Consortium, *Educational Object Economy*, At http://trp.research.apple.com.

15 Eisenberg, M.; Fischer, G., *Programmable Design Environments: Integrating End-User Programming with Domain-Oriented Assistance,* In Proceedings of Human Factors in Computing Systems, CHI'94, Boston, MA, pp. 431-437.

16 Engelbart, D. C.; English, W. K., *A Research Center for Augmenting Human Intellect* In *Proceedings of the AFIPS Fall Joint Computer Conference*; The Thompson Book Company: Washington, D.C., 1968; pp. 395-410.

17 Fischer, G., *Beyond Human Computer Interaction: Designing Useful and Usable Computational Environments* In *People and Computers VIII: Proceedings of the HCI'93 Conference (Loughborough, England)*; Cambridge University Press: Cambridge, UK, 1993; pp. 17-31.

18 Fischer, G., *Domain-Oriented Design Environments*, In *Automated Software Engineering* 1994, *1*, pp. 177-203.

19 Fischer, G., *Evolution of Complex Systems by Supporting Collaborating Communities of Practice,* In Proceedings of International Conference on Computers in Education, Kuching, Malaysi, pp. 9-17.

20 Fischer, G.; Lemke, A. C., *Construction Kits and Design Environments: Steps Toward Human Problem-Domain Communication*, In *Human-Computer Interaction* 1988, *3*, pp. 179-222.

21 Fischer, G.; McCall, R.; Ostwald, J.; Reeves, B.; Shipman, F., *Seeding, Evolutionary Growth and Reseeding: Supporting Incremental Development of Design Environments,* In Proceedings of Human Factors in Computing Systems (CHI'94), Boston, MA, pp. 292-298.

22 Fischer, G.; Nakakoji, K.; Ostwald, J.; Stahl, G.; Sumner, T., *Embedding Critics in Design Environments*, In *The Knowledge Engineering Review Journal* 1993, *8*, pp. 285-307.

23 Flanagan, J.; Huang, T.; Jones, P.; Kasif, S., "Human-Centered Systems: Information, Interactivity, and Intelligence," Beckman Institute for Advanced Science and Technology, 1977, Report# BOG-1, also available at: http://tortie.me.uiuc.edu/~pmj/hcs.html.

24 Gamelan, *The Offical Directory for Java*, At http://www.gamelan.com.

25 Girgensohn, A., *End-User Modifiability in Knowledge-Based Design Environments* Ph.D. Thesis, University of Colorado, 1992 (Also available as TechReport CU-CS-595-92).

26 Grudin, J., *Interactive Systems: Bridging the Gaps Between Developers and Users*, In *Computer* 1991, *24*, pp. 59-69.

27 Henderson, A.; Kyng, M., *There's No Place Like Home: Continuing Design in Use* In *Design at Work: Cooperative Design of Computer Systems*; J. Greenbaum and M. Kyng, Ed.; Lawrence Erlbaum Associates, Inc.: Hillsdale, NJ, 1991; pp. 219-240.

28 *Handbook of Human-Computer Interaction*; Hendler, M. G.; Landauer, T. K.; Prabhu, P. V., Ed.; Elsevier: Amsterdam, 1997, pages.

29 Illich, I., *Deschooling Society*; Harper and Row: New York, 1971.

30 Illich, I., *Tools for Conviviality*; Harper and Row: New York, 1973.

31 Kay, A. C., *Microelectronics and the Personal Computer* In *Scientific American*, 1977; pp. 231-244.

32 L3D, *Center for LifeLong Learning & Design Home Page*, At http://www.cs.colorado.edu/~l3d.

33 Landauer, T. K., *The Trouble with Computers*; MIT Press: Cambridge, MA, 1995.

34 Laurel, B., *Computers as Theatre*; Addison-Wesley Publishing Company: Reading, MA, 1991.

35 Mackay, W. E., *Triggers and Barriers to Customizing Software* In *Human Factors in Computing Systems, CHI'91 Conference Proceedings (New Orleans, LA)*, 1991; pp. 153-160.

36 McLuhan, M., *Understanding Media: The Extensions of Man*; The MIT Press: Cambridge, Massachusetts, 1964.

37 Nakakoji, K., *Increasing Shared Understanding of a Design Task Between Designers and Design Environments: The Role of a Specification Component* Ph.D. Thesis, University of Colorado at Boulder, 1993 (Also available as TechReport CU-CS-651-93).

38 Nakakoji, K.; Aoki, A.; Reeves, B. N., *Knowledge-Based Cognitive Support for Multimedia Information Design*, In *Information and Software Technology* 1996, *38*, pp. 191-200.

39 Nardi, B. A., *A Small Matter of Programming*; The MIT Press: Cambridge, MA, 1993.

40 Netscape, *Netscape Communicator Source Code*, At http://www.mozilla.org.

41 Norman, D. A., *Things That Make Us Smart*; Addison-Wesley Publishing Company: Reading, MA, 1993.

42 Norman, D. A.; Draper, S. W., *User Centered System Design, New Perspectives on Human-Computer Interaction*; Lawrence Erlbaum Associates, Inc.: Hillsdale, NJ, 1986.

43 Ostwald, J., *Dynasites*, At http://www.cs.colorado.edu-/~ostwald-/home.html, 1997.

44 Postman, N., *Amusing Ourselves to Death—Public Discourse in the Age of Show Business*; Penguin Books: New York, 1985.

45 Raymond, E. S., *The Cathedral and the Bazaar*, At http://earthspace.net/~esr/writings/cathedral-bazaar/cathedral-bazaar.html, 1998.

46 Repenning, A., *Programming Substrates to Create Interactive Learning Environments*, In *Journal of Interactive Learning Environments, Special Issue on End-User Environments* 1994, *4*, pp. 45-74.

47 Repenning, A.; Others, At http://www.cs.colorado.edu-/~l3d/systems/agentsheets/, 1997.

48 Resnick, M., *Turtles, Termites, and Traffic Jams*; The MIT Press: Cambridge, MA, 1994.

49 Rittel, H., *Second-Generation Design Methods* In *Developments in Design Methodology*; N. Cross, Ed.; John Wiley & Sons: New York, 1984; pp. 317-327.

50 Schön, D. A., *The Reflective Practitioner: How Professionals Think in Action*; Basic Books: New York, 1983.

51 Stefik, M. J., *The Next Knowledge Medium* In *AI Magazine*, 1986; Vol. 7; pp. 34-46.

52 Winograd, T.; Flores, F., *Understanding Computers and Cognition: A New Foundation for Design*; Ablex Publishing Corporation: Norwood, NJ, 1986.