

Review of *Computers and Design in Context*

for *User Modeling and User-Adapted Interaction*

An International Journal

Carol Strohecker

Strohecker, C. 2001. Review of *Computers and Design in Context*
(Kyng, M., and L. Mathiassen, L., eds., Cambridge, MA: MIT Press, 1977).
User Modeling and User-Adapted Interaction 11:3, 261-266.
Originally appeared as MERL Working Paper 2000-06.

Abstract

This is a review of an MIT Press book, *Computers and Design in Context*, edited by M. Kyng and L. Mathiassen (1997). The contributing authors shape an international research agenda based on conceptual tools forged through more than two decades of workplace computerization. The authors report attempts to extend old models such as end-user computing, end-user tailoring, participatory design, and speech act theory; to adopt new focuses such as on design rationale, the style of artifacts-in-use, control issues, aesthetic issues, ethical issues, self-representations of computational systems, and stakeholders' different views and expectations; to borrow from other domains such as jazz and art/craft traditions; to invent new approaches such as participatory evolutionary development, cooperative experimental system development, information systems research guidelines, communication strategies, and alliances with workers and worker associations; and to adapt new conceptual approaches such as ethnocriticism.

Review of Computers and Design in Context

M. Kyng and L. Mathiassen, editors
Cambridge, MA: MIT Press, 1997

Carol Strohecker, Sr. Research Scientist
MERL - Mitsubishi Electric Research Laboratory
201 Broadway, Cambridge, MA 02139 USA
01-617-621-7517 voice, 01-617-621-7550 fax
stro@merl.com
<http://www.merl.com/people/strohecker/index.html>

This book reads like therapy for shell-shocked system designers. One imagines the authors coming together to unload horror stories, soothe dashed hopes and damaged egos, and form solutions to complex issues that continue to haunt us as computers become ever more present in workplaces throughout the world.

Indeed, three conferences in Denmark led to the work reported in this 400-page volume. It includes papers selected from the 1995 “Computers in Context: Joining Forces in Design” conference, which combined concerns from two earlier meetings, the 1975 conference on “Working Practices in Systems Development” and the 1985 conference on “Development and Use of Computer-Based Systems and Tools – in the Context of Democratization of Work.”

The authors’ insights are enormously valuable. They shape an international research agenda based on whatever conceptual tools the system designers, developers, and users could forge or bring to bear on peculiar problems faced through more than two decades of workplace computerization. The authors report attempts to extend old models such as end-user computing, end-user tailoring, participatory design, and speech act theory; to adopt new focuses such as on design rationale, the style of artifacts-in-use, control issues, aesthetic issues, ethical issues, self-representations of computational systems, and stakeholders’ different views and expectations; to borrow from other domains such as jazz and art/craft traditions; to invent new approaches such as participatory evolutionary development, cooperative experimental system development, information systems research guidelines, communication strategies, and alliances with workers and worker associations; and to adapt new conceptual approaches such as ethnocriticism.

Editors Kyng and Mathiassen weave cacophony into chorus by organizing the book as two primary focuses: problems and themes in understanding artifacts and their employment in a context of use (chapters 1-5), and designing computer artifacts (chapters 6-14). We encounter throughout the characteristic emphases on the close relationship between design and use, and the themes of appropriateness, power shifts, and use-as-learning.

In Chapter 1, Sumner and Stolze present a case study and set of concepts for a new approach to software development. Participatory Evolutionary Development departs from the analysis-design-implementation sequence typical of in-house development, contractual development, and product development. Instead, users design and evolve their own tools, often incorporating off-

the-shelf software. In this “toolbelt” model, users, designers, and developers collaborate to shape better long-term work practices. Through a case study of the Voice Dialog Design system, the authors describe the importance of tailoring multiple representations such as flowcharts, tables, plans, and simulations to the needs of each of the major stakeholder groups. The authors see Participatory Evolutionary Development as the convergence of end-user computing and participatory design.

In Chapter 2, Beardon, Gollifer, Rose, and Worden ask how information system design practices and artist/craft traditions might inform one another. The domains are most comparable for situations in which people work alone, both at the point of production and in assuming responsibility for all aspects of a design’s life cycle. The authors identify concept formation and finishing as two stages of creative work and note general insufficiencies of computational tools in supporting these phases separately. A further problem is compounded by artists’ and designers’ awareness that software is a designed tool. The common wisdom that basing a user-interface metaphor the users’ domain of practice doesn’t work for users who are always analyzing and second-guessing aspects of a tool. Supplementing this critical awareness, artists deal with two simultaneous learning curves, one that pertains to generating their own creative productions, the other to “a maze of someone else’s making” – the user interface. Furthermore, the traditional manager-worker model of Participatory Design is difficult to apply in this domain, as artists tend to expect total control over their tools. Thus focusing on artists’ concerns leads to rethinking issues of responsibility, control, and power in system design.

In Chapter 3, Mørch describes a model for end-user tailoring of computer artifacts through techniques of customization, integration, and extension. Tailoring-as-design implies continued development in the course of ongoing modifications. Tailoring-as-use involves continued development in response to an application revealing some deficiency. In considering tailoring as an approach to helping users solve their own problems, Mørch urges consideration of how users can tailor without direct access to the original developers of the application, how tailoring can be better aligned with other work activities, and how can tailoring be a more integrated human activity, independent of specifically computational applications like spreadsheets and hypermedia.

In Chapter 4, Thoresen examines the role of design rationale in producing a range of technologies and examining their use in heterogenous settings. A key concern is how users see the design rationale reflected in the system and how interplay between the organizational context and the rationale contributes to different patterns of use and, ultimately, to opinions of the system. The author emphasizes complexities that emerge as a rationale, its implementation, and contextual conditions interact. In particular, users’ acceptance of a system’s usefulness is an essential condition for learning how to use it.

In Chapter 5, Ehn, Meggerle, Steen, and Svedemar define the style of artifacts-in-use as a balance between structure, function, and form. These concerns pertain respectively to technology, contextual social utility, and subjective experiences of using the system. The structure of an artifact concerns its material, or medium, aspect in the objective world. The artifact’s function derives from its implemented features. Form concerns the social, subjective world associated with the experience of use. The authors consider artifacts’ quality as emerging

from technical control but also from ethics and aesthetics. Control pertains to considerations like efficiency, error tolerance, consistency, robustness, maintainability, and portability. Ethics concern the question, “For whom?” and, more particularly, who loses and who wins. These are contextual concerns, often related to power. Aesthetics deals not just with the question of what is appealing or satisfying, but why. Ultimately, usability becomes a matter of appropriateness – a proper balance between structure, function, and form. This balance profoundly affects how people assess the artifact.

In Chapter 6, Dourish is concerned with self-representations of computational systems. He describes a tension between traditional process-oriented views of interface design and improvisation-oriented views of interface activity. Beyond the usual Participatory Design concerns of how we go about system design, he encourages reflection on what artifacts we use in design processes and what systems we choose to design. Resources that support improvised work at the interface can illuminate how users understand system activity, so designers can focus on how systems and devices find and present such information. Dourish introduces the notion of “accounts” as causally connected representations of system action that systems offer as explanations of their own activity. Accounts become a means by which users can match system functionality to work requirements. Accounts can also be a means of connecting insights from observational analysis to the practice of systems design.

In Chapter 7, Näslund presents a case study of an industrial development project, considering stakeholders’ different views. The author plays the interesting double-role of participant observer of issues related to usability, and observer-participant regarding other issues in the project. People concerned with the application placed it in different contexts, which made it possible for each actor to apply core expertise. Inevitably, the actors viewed the professional system differently, yet the differences were not identified within the development project and were not well understood overall. Each group assumed that the other actors had similar views, so misunderstandings and other communication problems emerged. Complicating the cross-talk were pressures to act according to specific views.

In Chapter 8, Grønbaek, Kyng, and Mogensen describe Cooperative Experimental System Development, which entails active user involvement throughout an entire system development process; prototyping experiments that are closely coupled to work situations and use scenarios; early, cooperative analysis and design results transformed to object-oriented design, specification, and realization; and design for tailorability. CESD supports a cooperative learning process with common learning agendas for practitioners from the user organization and system developers. Compared to established participatory design and cooperative design approaches, developers cooperate throughout the course of a CESD process. They focus on transforming loosely specified design artifacts such as mock-ups and prototypes into properly engineered and documented computer systems.

In Chapter 9, McMaster, Jones, and Wood-Harper examine the failure of a prize-winning parking control system to meet users’ expectations. This paper includes one of the highlights of the book, a wonderful cartoon that whimsically illustrates the players’ conflicting understandings and expectations of the computational system. In hindsight, the authors advocate for development of communication strategies to engineer stakeholder expectations. The goal is to reduce the gap

between designers' assumptions of stakeholder attitudes about the system, and stakeholders' actual feelings. The strategies should promote benefits of the system in an appropriate and timely manner, in order to allay any fears and concerns held by stakeholders. This process necessarily entails discussion and collaboration. Communication strategists need to develop content, means of delivering information to the target group, and a time frame for its delivery.

In Chapter 10, Blomberg, Suchman, and Trigg emphasize developers' need for awareness of users' organizational context and the need for strengthening ties between sites of system design and use. The authors clarify that their own work context combines the roles of technology producer and employer of system developers. Desires to cut labor costs can motivate these players' employment but negatively impact the interests of other players as managers resort to automation initiatives, use of contingent labor, and outsourcing. The authors advocate alliance with workers and worker associations who are affected by current or proposed new technologies and who want to gain increased influence over technology development and related organizational changes in their workplace. Developing case-based prototypes is one system-design technique that can help to address such concerns.

In Chapter 11, Bratteteig and Stolterman consider the jazz group as a metaphor for collective work practice and innovation. Desirable qualities of group design processes include creating visions and new ideas, formulating specifications on the basis of a range of competencies, and cultivating mutual learning and understanding. Orchestration of a design group needs to support the group's imaginative thinking, synthesize competence, and provide managerial support for productions. Traditionally, system development methods tend to be based on the view that the way a development process is carried out influences the quality of the resulting computer-based system. Methods control costs, time, and people, and emphasize documentation of development processes. However, such intense control of system designers may undermine their ability to create something novel. By orchestrating a group as though arranging a jazz improvisation, rationalistic methods can be supplemented with methods, techniques, and managerial support that activates and balances a range of rationalities in the design process.

In Chapter 12, Ljungberg and Holm recommend evaluating a development situation in order to determine the appropriateness of using speech act theory in design processes. System developers often adapt this theory when applying it to a new field, but the authors call attention to the need for particular adaptations, such as focusing on relations between organizational commitments, elaborating the notion of organizational commitments (as opposed to those of individuals), discussing alternative classification criteria for design, and possibly extending the speech act concept with a typology of specific situation types.

In Chapter 13, Muller elucidates heuristics to help HCI professionals deal with multiple voices and heterogenous perspectives. He finds ethnocriticism useful in promoting collaboration and dialog across differences in culture and power. He explains three components: a pedagogical view, which deals with multiculturalism; a socio-political view, which deals with polyvocal polity; and an ethical view, which deals with heterogeneity. Considering HCI from these viewpoints can help in going beyond modernist assumptions that privilege logic, rationality, and efficiency, to focusing on users' motivations, goals, and experiences. Ultimately ethnocriticism can lead toward more ethical HCI practice. The author includes a helpful glossary of terms,

including “postmodernism,” “appropriation,” “privilege,” and “narrativization.” The paper concludes with a refreshing self-critique that poses the question of whether Bakhtin’s concept of dialoguism can lead to a notion of a polyvocal paper and/or other new styles for technical publications.

In Chapter 14, Braa and Vidgen describe how the Information Systems framework for managing research on construction and use of technical artifacts in an organizational context. The framework includes design guidelines considering both the research issue and any organizational constraints, controls for the project as the research takes place in the organizational laboratory, and a basis for reconciling the inquiry with the research plan. Information Systems researchers approach their work through reduction, interpretation, and intervention, and categorize results according to prediction, understanding, and change.

Researchers in user-modeling and user-adapted interaction will benefit from noting the collective emphasis on work preceding design and implementation to find out who the users are, what their work entails, what tools they prefer, and so on. Of course the parallel emphasis is on asking users to report on themselves and their own practices, and the book becomes a manual for how to do the asking.

Biography:

Carol Strohecker is a Senior Research Scientist with Mitsubishi Electric Research Laboratory in Cambridge, Massachusetts (MERL). She is concerned with how people learn and how objects, artifacts, and technologies can facilitate learning. Her designs for computational media support human cognition and expression while enabling study of these processes. Research interests include interactive narratives, virtual construction kits embodying fundamental principles of math and science, and environments that support constructive activities in complementary physical and virtual domains.

Dr. Strohecker is a Presidential Nominee on the MIT Corporation Visiting Committee for the Department of Architecture and Media Arts and Sciences. She is on the advisory board of the NSF-funded “Making Models” programs and exhibits at Boston’s Museum of Science. Dr. Strohecker serves as a reviewer for several conferences, publications, and academic institutions, and has been a Fellow of the Harvard University Graduate School of Design, the Massachusetts Council for the Arts and Humanities, and the U.S. National Endowment for the Arts. She completed her graduate study at the MIT Media Laboratory in 1991.