

# Contextual Design: An Emergent View of System Design

*Dennis Wixon, Karen Holtzblatt*

*Stephen Knox*

Digital Equipment Corporation  
110 Spit Brook Rd (ZKO2-3/K06)  
Nashua, NH 03062-2698

1-603-881-2276, Wixon%kobal.dec@decwrl.dec.com

## ABSTRACT

We offer an introduction to contextual design as an emergent method for building effective systems. Contextual design addresses a number of the inadequacies in previous methods by emphasizing: interview methods conducted in the context of the user's work, codesigning with the user, building an understanding of work in context, and summarizing conclusions through out the research. We contrast this design method to usability engineering and artifact examination.

**KEYWORDS:** theory, design, methodology

## OVERVIEW

This paper communicates our position on the theory and practice of human factors in system design. We are particularly motivated by Carroll and Kellogg's [2] excellent article calling for a study of artifacts to derive theories for interface design. While we support their effort at formalizing a method of design, we question their statements about hermeneutic design approaches. In particular, we make a case for the relevance of one hermeneutic design technique, contextual design, to engineering practice. This method was developed to design usable products. It results in an understanding of customer work which can be generalized to new design problems.

To make our case, we will document a growing belief

in the Human-Factors/HCI community that traditional, predictive research techniques have fallen short of delivering timely and relevant design information. We will outline the procedures of contextual design (CD). We will also contrast contextual design to the procedures described by Carroll and Kellogg, noting that the major differences may be a result of different goals: product design versus theory development.

## THE CHALLENGE TO BUILD EFFECTIVE SYSTEMS AND SOME RESPONSES

### Changing Perspectives in CHI and Inadequacy of Traditional Methods

At the 1989 CHI conference, we noted the several panel sessions devoted to non-traditional design techniques, such as "Contextual Field Research" and "Drama and Personality in User Interface Design". We also witnessed the tension between traditional and non-traditional views in the panel session "The Role of Laboratory Experiments in HCI: Help, Hindrance, or Ho-Hum?". This tension is at the heart of the Carroll and Kellogg paper, which appears motivated by a desire to synthesize hermeneutic approaches and those predicated on quantitative models of human performance.

We contend that this tension derives from a general recognition that traditional methods have failed to provide relevant information for a number of necessary elements of product development such as: needs analysis, requirements definition, and interface design. This inadequacy is keenly felt in the industrial environment, where technological developments are rapidly accelerating. The need to respond with timely, informed, and realistic designs (implementable against a background of limited resources) is paramount.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish requires a fee and/or specific permission.

Some investigators claim that the breakdown in traditional human factors practice has resulted from an inability to integrate findings of independent laboratory studies. Their solution is to build a unified, comprehensive model of human performance either by developing more complex experimental paradigms [6, 26] or by integrating experimental results into a single framework [14, 27].

We agree with another set of investigators, who see the problem as the difficulty in applying laboratory findings to the design of real world systems. [28, 1, 18, 24, 25, 30, 3]. We further contend that the root cause of this problem is the method itself. We claim that the practice of traditional human factors to address concerns of design is a misapplication of experimentation borrowed from the physical sciences.

Many human factors professionals come to industry steeped in the methodology of experimental psychology or industrial engineering. This methodology was not developed for product design. It has become, by default, the dominant methodological paradigm in the human factors profession primarily because of early successes in establishing relationships between perceptual/motor performance and system engineering. However, the challenges of interface design have changed dramatically in the last 25 years.

#### Responding to the Challenge: the Development of New Techniques

Today, design engineers are under intense pressure to design effective interfaces and thus must creatively synthesize and integrate available technologies into a unified solution. The questions faced are not adequately addressed by quantitative models predicting how fast subjects can perform benchmark tasks or the decrease in error rates from method A to method B. The HCI professional often need to develop new, non-traditional techniques that yield an *understanding* of real customers solving real problems in the real world [10].

Many of the non-traditional techniques utilize once discounted verbal data collected in controlled, laboratory-like environments [16, 15, 7]. Contextual design approaches go beyond verbal data by incorporating observations made in the user's context. These observations are not bounded by techniques which pre-define categories of observation and subsequently analyze "results". Rather, the value in the data comes from interpreting what is observed in its context and by engaging the customer in a dialog about these interpretations [30]. Such techniques are increasingly

practiced by the engineering community. For example, at Digital these techniques were used on products such as XUI, DECwrite, DECdecision, EVE, and VAX Rally 2.0. In the industry as a whole they have been used at IBM, Tectronix, Pacific Bell, MCC, Candle, and the University of York [8]. Contextual design methods are chosen because they pass the acid test: providing timely, usable information that aids in the making of critical design decisions.

#### Comparisons between the non-traditional techniques

A number of new approaches to user interface design are being clarified and articulated. Some examples are usability engineering (UE) [29] contextual design (CD) [30, 29] and the approach advocated by Carroll and Kellogg, which might be called artifact examination (AE). These approaches differ in fundamental assumptions, choice of methods, overall goals, and interpretation of results. At the same time the approaches overlap to a varying extent.

A simplistic summary of the differences in these approaches is captured in Table 1.

Both usability engineering and the contextual design place a greater emphasis on understanding user needs in context than artifact examination. Usability engineering differs from contextual design in its use of operational specifications to focus the usability effort. Artifact examination uses a generic task analysis which is absent from other approaches. Since contextual design is the topic of this paper and since it may be less well understood than the other approaches, the following section is aimed at briefly describing this approach.

## CONTEXTUAL DESIGN

### The Role of Contextual Design in Developing Products

Contextual design as developed and practiced by us is intended to support the development of products. Although the methods used in contextual design were adapted from various field research techniques, its primary purpose is not to develop theory.

Our challenge within Digital is to design products in a cost effective and timely fashion that supports, extends and transforms user work. There are two critical phrases in the statement of our intention: "transform user work", and "cost effective" design.

To "transform user work" we must understand: what that work is, what kind of computer system would support that work, how technology could transform such work, and how to anticipate and support the changes in work that new technology brings. Our assumption is that systems are designed from a set of explicit (or implicit) assumptions about how users work.

For example, in VAX RALLY V1.0 software, the system provided menus of variable names which eliminated recall and typing. However, in the part of the system, where the users did coding, no menus were provided. Designers assumed the users of the system would not do much coding and that, when coding, only a few variable names would be used, which would be easy to remember. Our studies showed that many users program and that they found recalling and correctly typing variable names difficult. It was clear that the product could be improved and user errors eliminated if users could pick variable names from a menu when coding. This feature was added in V2.0. Thus, version 2.0 of VAX RALLY better reflected the type of work users did. For us, a major task of design is to design a system that will fit with and enhance the user's existing way of working.

In contextual design we use contextual inquiry to yield data on user work as a basis for design. The cornerstone of contextual inquiry is to collect data in the context of users' work as a partner with users. We talk with users while they are working. We see which aspects of a system support or interfere with work and how that system interacts with the whole work context. If work occurs over months or hours, we track the work by talking about artifacts users have produced and events that occurred during the work. Different users of a system are selected so that each user adds another dimension to the emerging picture of

work breakdowns and opportunities to support work within the system to be developed.

Because we engage the users as partners in this inquiry, we gather users' interpretations, language, and the structuring of their work activities. The structure of users' work in many ways is revealed in users' thoughts, intentions and how they orient to their work tasks. Thus, talking with users to develop a shared understanding of these thoughts is key to uncovering the data needed to develop a usable system: a workflow and task concept which builds upon the users' existing decomposition of work.

The second critical phrase in our intention is to provide "cost effective and timely" information for design. When developing a product which requires data with respect to user work the relevant questions are: "How long will it take to get this data?" and "How much data do I need?"

Because of aggressive development schedules, the design will be often done with a small sample of users. In contextual design we accept this constraint and seek to maximize differences between interviewees to get a comprehensive view of user work with as few participants as possible. Moreover we build understandings at the same time that data are collected. Thus, results are available at any point in the inquiry. There is no need to wait for the "end of a study" or "enough users".

#### Reusable Knowledge and Contextual Design

If an understanding of each product domain, or worse, each product must be developed from scratch as though no prior knowledge is relevant, we would not be cost effective. Implicit in any good methodology, for science or engineering, is that it produces knowledge which can be used in new situations.

<b>AE: Artifact + generic task analysis</b>	<b>-&gt; theory</b>	<b>-&gt; Design</b>
<b>UE: User work + technology</b>	<b>-&gt; usability specification</b>	<b>-&gt; Design</b>
<b>CD: Work in context + interpretation</b>	<b>-&gt; understanding</b>	<b>-&gt; Design</b>
	<b>+ technology</b>	

**Table 1: Comparison of Non-traditional methods**

Contextual inquiry creates an understanding of user work and usability which informs engineers while they design. While this understanding is relevant to a particular work domain and thus to any tools within this work domain, it also suggests generic principles of usability and work concepts that become the initial framework for the design of new products. This initial framework must then be iterated or shaped by domain-specific information.

For example, in one domain, a user of a word processing system performed production work transcribing reports for a hospital. The reports varied by format and paper color for printing. As a result, she organized her work into groups using the same format and paper color. Rather than making each report a separate file within each format group, the user put more than one report in a file. But this strategy necessitated manual pagination which required her to be very vigilant to assure correct page numbers. Although the product provided a pagination facility, it was designed assuming that each file represented a single continuous document. In addition the system did not provide a way to easily copy a format and enter new text. Thus, the user developed a workaround to suit her needs. As she said "It's as if the product has a mind and is saying this is how you do it. But that's not how I wanted to do it... It didn't really fit my need."

Understanding work disruption in one context we were able to use it in other design efforts. For example, in the Digital toolkit for X Window System, XUI, we developed a number of software routines (widgets and gadgets) to support specific user actions such as the selection of existing files, the storage of new files, and others. These widgets allow users to access and store information without leaving an application and disrupting the flow of work. Thus, the general principle of maintaining the flow of work generated the rationale for the design of a specific interface component. The component is then instantiated in code, which is shared among a whole family of products. In this way general principles do more than give us a beginning understanding; they can actually shorten the time it takes to develop products.

Each set of principles from contextual interviews informs the data-gathering in a new application area. Although the concepts start out domain-specific, they evolve as designers move from domain to domain. Within specific areas work with users refines and develops the concepts appropriate to the new domain. Our concepts of work and usability grow from product to product. We never start from scratch; we always build upon our growing knowledge.

Because we recognize that we carry this framework or focus around with us all the time, contextual design builds into its methodology a way of challenging and evolving this focus to ensure that we do not over-generalize. We must validate these frameworks with users within a specific domain. We do this by continuously sharing our assumptions and interpretations with users and by inviting them to interpret and design with us.

### The Contextual Inquiry Process

We stated earlier that contextual inquiry was developed for the explicit purpose of supporting engineering efforts, not scientific efforts. Necessarily, contextual inquiry uses a well-defined set of procedures. These procedures differ from traditional experimental techniques, and have not been systematically and completely presented as a methodology in the public literature. As Carroll and Kellogg [2, p. 7] point out we have offered "no systematic methodology". We are now in the process of training our engineers to do contextual inquiry including the interviewing, analysis and concept development. We have found our procedures to be communicable. Today the procedure may be summarized as follows:

- Interview users about their work in the real work environment
- Be concrete, talk about what the user is doing, just did, or talk in the context of a work product
- Let the user lead the conversation, co-interpret and co-design: share your understandings of their work and design ideas with users to validate and to stimulate user reflectivity, inquire about their assumptions
- Expand and challenge the background of assumptions one brings to each interview, probe all surprises, and assumptions. Share your assumptions with users. Pick users strategically so that they are representative of the diversity of your user base.
- Summarize your understanding at the end of each session to determine who to talk to next and what to focus on next.
- Build an understanding of user work and of the environment to be designed.
- Based on this understand, design and build a first cut at a User Environment.

- Iterate the prototype with real users doing real work (if possible) and evaluate with contextual inquiry.

Practically these steps need not be serial. Also, specific methods such as video recording, audio taping, or note taking during are used as a matter of convenience or personal style.

### CONTRASTS BETWEEN CONTEXTUAL DESIGN AND ARTIFACT EXAMINATION

In this section contextual design is compared with the methods Carroll and Kellogg suggest, which we term Artifact Examination.

#### Design principles are inherent in objects

A major claim and important insight of artifact examination is that the usability of an object is determined "by independent psychological rationales inherent in its design". While true, we ask the question: "Is the resulting design a good one?" While there may be principles inherent in the design of objects, these principles could have produced an unusable product. As Norman [19] points out, many products are poorly designed. Examining such artifacts for their "inherent design" principles could lead to more poor designs. In contextual design, we place a strong emphasis on studying people's work and the use of products in that work.

In addition, any inherent psychological principles in a design may be difficult to ferret out or are simply not present. The design of a product is the result of a number of competing forces: resources, features, time to market, and competitive position. These forces have often overwhelmed usability issues in design. Examination of a product which was designed exclusively to minimize time to market and cost might reveal few usability principles. In contextual design we rely on the study of products in the context of use to ferret out the effective and ineffective aspects of the design.

We don't really believe that Carroll and Kellogg intend to extract design principles solely by examining artifacts in isolation. It is clear Carroll and Kellogg carefully studied the use of the training wheels editor and HyperCard. In the case of the training wheels system, [4, 5] the methodology employed involved examining the problems people had when using traditional documentation and producing prototypes which were improved through an empirical approach. The HyperCard case is less well documented, but it is fairly clear from their writing that Carroll and Kellogg

have either used HyperCard extensively or interviewed people who have. A clarification of how artifact examination uses interviews, reflections, or field data could help to reconcile apparent differences between artifact examination and contextual design.

#### Details of Design

As proponents of contextual design methods, we argue that principles alone are not enough in the design of artifacts. Design involves not only a broad architecture but also a myriad of details which are not obvious until observed in practice.

Once articulated as a paper specification, a simulation or as a functional prototype, a design can be evaluated. This can take place in the field or in the laboratory; the overriding concern being the extrapolation from the test situations to the actual situation of use. One can draw an analogy from psychological research. The relation between iterative testing and original user interface is somewhat similar to the relation between theory and experiment in traditional science. The overall experimental context is the product of a widely shared but often unstated matrix of assumptions about the definition and development of knowledge, and the particular variables manipulated, which are products of the specific theory being tested. Similarly, in interface design the overall design (theory) produces a particular interface (experiment) which is then evaluated in the context of use (experimental findings). The findings of these tests are used to alter both the overall design (theory) and the particular details of the implementation (experiment).

Clearly, there are important differences between a user interface prototype and an experiment. One set of differences arises from practical concerns such as the time critical nature of results for product development, or the need to collect, organize, and present results in a way that they can be most readily assimilated by a development group. A second and more interesting set of concerns are philosophical. Some examples would be a commitment to involve the user as a partner in design, and a greater emphasis on discovery of principles as compared to the testing of alternatives.

#### Role of falsifiable claims

Artifact examination places a strong emphasis on falsifiable claims. This concern for falsifiable claims is rooted in a particular view of science which claims that a meaningful hypothesis is one that can be disconfirmed [23]. From the viewpoint of contextual design, we interpret this requirement differently than the

classical view.

For example, design principle like "Users want and need to be in control of the tools they use to perform their work" [33]. One can imagine ways to disconfirm this hypothesis using traditional laboratory methods. In contrast in designing an interface, we seek to develop with users an understanding of this principle in the context of their work. By refining what "control" means in the context of a specific product we can produce an effective design.

The design principles offered by major software/hardware vendors [12, 21, 20, 13] are quite similar. We expect most vendors are working on refining the interpretation of these principles rather than seeking to disconfirm them using classical experiments.

### Artifacts as a source of design

The use of artifacts to create new designs is a time-honored tradition and is widely used. Certainly design is often or can be seen as an evolutionary process building on designs of the past. However, design also has an additional aspect. Good design breaks new ground, applies existing technology to new domains, integrates features from diverse sources into a single design, exploits on new technology, or extends existing technology beyond limits. Pelle Ehn [8] considers design as a dialectic between tradition (analysis of existing artifacts and practices) and transcendence (new directions).

In contextual design, we seek a wide range of sources for inspiration. We see spreadsheets and statistical analysis packages as examples to be emulated. The designs for these did not derive from existing software artifacts, rather they represent automated embodiments of clearly specified procedures, i.e. institutionalized social practices. These procedures were extremely labor intensive for people but were trivial for machines. The design of products involves more than the examination of existing products for principles. It also involves the examination of unaided procedures as candidates for automation.

### Mystical design decisions

Artifact examination claims that, in the hermeneutic approach to design, decision making is "essentially mystical". While elements of mysticism are often involved in designing usable systems, an essentially mystic approach would hardly be effective in an engineering environment.

Pepper (1966) says that mysticism is rooted in the

mystic experience which is revelatory, immediate, indubitable, ecstatic, unified, comprehensive, and negating other modes of cognition. We have all had experiences like this to varying degrees, in art appreciation, the natural world, or in love. We may have had this experience in design, an aspect of a product may "clearly feel right" or "clearly feel wrong". However, the intensity of our revelation is not usually adequate to convince software developers to change designs or managers to allocate resources. To effect design change we must work effectively with others who may or may not share our experience. We can't afford to become hermits as many mystics do. Instead we focus on developing a shared understanding of the experience of the user. The aim of contextual inquiry is to make the user's experience public so it can drive the social process of design.

### CONCLUSION

We have summarized what we see as a breakdown in traditional methods of design and applied research. We have briefly sketched contextual design. We have placed it in its origins in the product development process and outlined its methods, particularly the contextual interview. This approach has its roots in the work of Pelle Ehn [8], Winograd and Flores [31], Heidegger [10], Wittgenstein, [32], and Marx [17]. We have contrasted this approach with that of artifact examination.

We sense that practitioners of product design are developing new and effective methods. Our belief is that the presentation and documentation of these methods will lead to their refinement. Contextual design is an evolving collection of methods. We continue integration of methods and synergy between diverse approaches.

### NOTES

The views expressed are those of the authors and do not necessarily represent the views of Digital Equipment Corporation.

The authors thank: Alana Brassard, Anne Duncan, Charles Frean, Michael Good, Derek Kozikowski, Dave Marca, and Chauncey Wilson for their careful reviews. We also acknowledge John Whiteside for his knowledge and guidance.

XUI, DECdecision, DECwrite, VAX, RALLY are trademarks of Digital Equipment Corporation, Maynard Mass.

IBM is a trademark of IBM corporation.

Apple, HyperCard, are registered trademarks of Apple Computer Inc.

OPEN LOOK is a trademark of AT&T.

OSF/Motif is a trademark of Open Software Foundation, Inc.

X Window System is a trademark of the Massachusetts Institute of Technology.

## REFERENCES

1. Boff, K.R. (1988). The value of research is in the eye of the beholder. *Human Factors Society Bulletin*, 31(6), 1-4.
2. Carroll, J.M. and Kellogg, W.A. (1989). Artifact as theory nexus: Hermeneutics meets theory-based design. *Proceedings of the Conference of Human Factors in Computing systems*. Austin, 7-14.
3. Carroll, J.M. and Campbell, R.L. (1986). Softening-up hard science: Reply to Newell and Card. *Human-Computer Interaction*, 2, 227-249.
4. Carroll, J.M. and Carrithers, C. (1984). Blocking errors in a training wheels system. *Human Factors*, 26,, 377-389.
5. Catrambone, R. and Carroll, J. M. (1987). Learning a word processing systems with training wheels and guided exploration. In J.M. Carroll and P.P. Tanner, (Eds.) *Proceeding of CHI + GI'87: Conference on Human Factors in Computing and Graphics Interface*. New York.: ACM papers 169-173.
6. Charlton, S.G. (1988). An edpidemiological approach to the criteria gap in human factors engineering. *Human Factors Society Bulletin*, 31(3), 1-3.
7. Connally, C.E. and Tullis, T.S. (1986). Evaluating the user interface: Videotaping without a camera. *Proceedings of The Human Factors Society, 30th Annual Meeting*, Santa Monica, CA: Human Factors Society, 1029-1033.
8. Ehn, P. *Work-Oriented Design of Computer Artifacts*, Stockholm: Arbetlivscentrum, 1988.
9. Good, M. (1989) Seven Experiences with Contextual Field Research. *SIGCHI Bulletin*. 20(4), 25-33.
10. Gould, J.D. and Lewis, C. (1985). Designing for Usability: Key principles and what designers think. *Communications of the ACM*, 28(3), 300-311.
11. Heidegger, M. *Being And Time*. (Trans. J Macquarrie, and E. Robinson). New York: Harper and Row, 1962.
12. *Human Interface Guidelines: Apple Desktop Interface*. (1987). New York: Addison Wesley.
13. *IBM Common User Access Advanced Interface Design Guide*. (1989), Boca Raton, FL: IBM
14. John, B.E. and Newell, A. (1989). Cummulating the science of HCI: From S-R compatibility to transcription typing. *Proceedings of the Conference of Human Factors in Computing Systems*, Austin, 109-114.
15. Kato, T. (1986) What "question-asking protocols" can say about the user interface. *International Journal Of Man-Machine Studies*, 25, 659-673.
16. Knox, S.T., Bailey, W.A. and Lynch, E.F. (1989). Directed Dialogue Protocols: Verbal data for user interface design. *Proceedings of the Conference of Human Factors in Computing Systems*, Austin, 283-287.
17. Marx, K. *Thesis On Feuerbach*. (Trans. T.B. Bottomore). New York: Penquin Books, 1971.
18. Meister, D. (1988). References to reality in the practice of human factors. *Human Factors Society Bulletin*, 31(10), 1-3.
19. Norman, D. *The Psychology Of Everyday Things*. New York, Basic books. 1988.
20. *Open Look Graphical User Interface Functional Specification*. Sun Microsystems: Mountain View, CA.
21. *OSF/MOTIF Style Guide* (1989). Open Software Foundation. Cambridge MA.
22. Pepper, S.C. (1966). *World Hypotheses*. Los Angeles: University of California Press.
23. Popper, K.R. (1959). *The Logic Of Discovery*. London: Hutchinson.
24. Rosson, M.B., Maass, S and Kellogg, W.A. (1988). The designer as user: Building requirements for design tools from design practice. *Communications OfThe ACM*, 31(11), 1288-1298.
25. Rouse, W.B. (1987). Much ado about data. *Human Factors Society Bulletin*, 30(9), 1-3.
26. Simon, C.W. (1987). Will egg-sucking ever be-

- come a science? *Human Factors Society Bulletin*, 30(6), 1-4
27. Smith, L.L. (1987). Whyfore human factors? *Human Factors Society Bulletin*, 30(2), 6-7.
28. Thomas, J.C. and Kellogg, W.A. (1989). Minimizing ecological gaps in interface design. *IEEE Software*, January, 78-86.
29. Whiteside, J., Bennett, J. & Holtzblatt, K. Usability engineering our experience and evolution. in Helander, M. (Ed.) *Handbook Of Human Computer Interaction*. New York: North Holland, 791-817.
30. Whiteside, J. and Wixon, D. (1987). Improving human-computer interaction -- a quest for cognitive science. In J.M. Carroll (Ed.) *Interfacing Thought: Cognitive Aspects Of Human Computer Interaction*. Cambridge, MA: Bradford/MIT Press, pages 337-352.
31. Winograd, T. and Flores, F. *Understanding Computers And Cognition*. Norwood, NJ: Ablex, 1986.
32. Wittgenstein, L. *Philosophical Investigations*, (Trans. G.E.M. Anscombe). New York: Macmillan, 1968.
33. *XUI Style Guide* (1988). Digital Equipment Corporation. Maynard MA.