MUST

A Method for Participatory Design

Kensing, Finn; Simonsen, Jesper; Bødker, Keld

Published in:
Human-Computer Interaction

DOI:
10.1207/s15327051hci1302_3

Publication date:
1998

Document Version
Peer reviewed version

Citation for published version (APA):
Kensing, F., Simonsen, J., & Bødker, K. (1998). MUST: A Method for Participatory Design. *Human-Computer Interaction*, *13*(2), 167-198. https://doi.org/10.1207/s15327051hci1302_3

MUST - A Method for Participatory Design

Finn Kensing, Jesper Simonsen, and Keld Bødker

Department of Computer Science
Roskilde University
P.O. Box 260, DK-4000 Roskilde
Denmark
Tel: +45 46 74 20 00
Fax: +45 46 74 30 72
e-mail: { kensing, simonsen, keldb}@ruc.dk

ABSTRACT
The article presents a conceptual framework and a coherent method for design in an organizational context within the participatory design tradition. The MUST method has been developed throughout 10 projects in Danish and American organizations, and it has recently been evaluated and adopted by 3 Danish organizations. The method is based on thorough participation with users and managers, and it combines the use of ethnographic techniques and intervention. The article describes the application area and perspective of the method, presents 6 general principles on which the method is based, and describes 5 main activities providing a stepwise decision-making process in the overall design process. Each of the main activities are illustrated by an example taken from our last project. The article concludes by summing up the main points.

 This article is originally published in **Human-Computer Interaction, Vol. 13, No. 2, 1998, pp. 167-198.**

CONTENTS

1. INTRODUCTION

The purpose of this article is to present a coherent method, the MUST[1] method, for participatory design (PD). Referring to the distinctions of Grudin (1991), the article presents a method for PD in an organizational context, where this context is either in-house/custom development or competitive bid/contract development. The method is not intended for Grudin's third category, the design of generic products for a large market.

We use the term *design* in the same way as architects do - focusing on the analysis of needs and opportunities and the design of functionality and form. We do acknowledge, however, that in a succeeding development process, further design is needed, and that when applying a computer system, users might very well find new ways of utilizing the system, as well as come up with additional demands. This does not negate the need for a design that is a good first approximation.

The MUST method has been developed throughout 10 projects in Danish and American organizations (Kensing, Bødker, & Simonsen, 1994), and it has recently been evaluated and adopted by information technology (IT) professionals within three Danish organizations, one of which is documented in Kensing, Simonsen, and Bødker (1998). The method is inspired by ethnographic approaches, (see, e.g., Blomberg et al., 1993; Blomberg, Suchman, & Trigg, 1996; Hughes, Randall, & Shapiro, 1992, 1993; Suchman, 1995), and by Scandinavian participatory design approaches, (see, e.g., Greenbaum & Kyng, 1991; Grønbæk, Kyng, & Mogensen, 1993; Kensing & Munk-Madsen, 1993; Kyng, 1995). We have designed IT support for 9 people on an editorial board of a film company and for 50 people working in a research and development lab; we have designed multimedia support for 140 people working at a radio station. All the work domains can be characterized as professional work in complex settings with a very open-ended agenda for the design project - no clear statement of the problems, of the kind of IT support needed, or of how the project should be carried out.

The MUST method relates to other approaches to design in a number of ways.

First, it is important to note that we offer a *conceptual framework of the design process*, whereas, for example, Stolterman (1991), and Ehn, Meggerle, Steen, and Svedemar (1997) developed a conceptual framework facilitating an

---

[1] MUST is a Danish acronym for theories of and methods for design activities.

ongoing evaluation of the qualities of the designed products. Yet others (e.g., Carstensen, 1996; Rasmussen, Pejtersen, & Goodstein, 1994; Schmidt, 1990, offered a framework for the conceptualization of users' work domain.

A second distinction is related to the application area of the proposed methods. The MUST method focuses on the *early activities in a development process,* like most PD methods, business process reengineering (BPR; Hammer & Champy, 1993) and object oriented analysis (Coad & Yourdon, 1991). It offers *guidelines for project management (like BPR) as well as for the design proper* (like PD and object oriented analysis). The Cooperative Experimental System Development method (Grønbæk, Kyng, & Mogensen, 1997) deals with the entire development process, but focuses solely on the design proper.

A third distinction, also related to the application area, is what kind of changes the design process strives towards. Although downsizing is an inevitable consequence of BPR (Hammer & Champy, 1993, p. 212), ethical issues in relation to involving users are not dealt with. We state explicitly that if management aims at job cuts or other *drastic changes*, this *should be announced up front*.. If users know and accept these objectives, we still recommend a participatory approach.

Fourth, the MUST method includes *management issues* in relation to design processes in an organizational context. This has not been dealt with earlier in PD literature, where the focus has been on why and how to work with users. Although S. Bødker (1996) reported on the role of management in relation to the future use of a system, she does not deal with the role of management in the processes of generating visions and helping them to materialize.

Fifth, though linking - or aligning - a business strategy to IT is addressed in information systems (IS) literature, especially within the field of strategic information systems planning, it is generally viewed as a (top) management issue (Cash, McFarlan, & McKenney, 1992; Keen, 1991; Lederer & Sethi, 1991) dominated by a rational top-down approach (Henderson & Sifonis, 1988; Premkumar & King, 1994; Yetton, Craig, & Johnson, 1995). Instead, in line with, for example, "the double-loop transformation process within strategic alignment" (Henderson & Venkatraman, 1992) and "the organizational approach" (Earl, 1993), we consider the *relations between a design project and an organization's business and IT strategies*. And we deal with these issues *from the perspective of IT professionals.*

As a sixth distinction, we argue for the need for a *separate design activity* including the development of visions for the overall change, to produce a sustainable basis for further development and implementation. Other approaches (see, e.g., Grønbæk et al., 1997) primarily strive for an accountable design through extensive user participation in prototyping.

Seventh, Plowman, Rogers, and Ramage (1995) reported that the dominant approach in projects using ethnography consists of sociologists conducting the ethnographic studies and informing IT professionals of their findings. We and others (see Blomberg et al., 1996; Shapiro, 1994) are working toward a closer link between ethnography and design. We recommend that *IT professionals start practicing ethnographic techniques* themselves in their cooperation with users. In some countries, such as Denmark, there is no tradition - in organizational settings - for involving sociologists or anthropologists in design projects. Our experiences confirm that it is possible and valuable for IT professionals to use ethnographic techniques as part of their design activities (K. Bødker & Kensing, 1994; Kensing & Winograd, 1991; Simonsen & Kensing, 1994, 1997, 1998).

Finally, although systems development methods suggest various formalisms for describing users' current work and the envisioned design of IT systems (e.g., Coad & Yourdon, 1991), formalisms play a minor role in the MUST method. Instead we suggest plain text, freehand drawings, and sketches for the production and presentation of the relation between proposed IT systems and users' current and future work practice. An extended use of formalisms is postponed until later on in the development process.

According to Mathiassen (1981), a method is characterized by its application area, its perspective, and its guidelines: techniques, representation tools, and principles for organizing a project. Our suggestions according to these categories are described later. Section 2 explains why IT professionals need a method like MUST as a resource for design activities. Section 3 presents and argues for the method's guiding principles, whereas Section 4 describes the main activities that constitute a design process according to MUST. We suggest techniques for each of the main activities and illustrate how they were carried out in a recent project. The article is concluded by a summary. For further examples, we refer the reader to K. Bødker (1990); K. Bødker and Kensing (1994); Kensing et al. (1998); Kensing, Bødker, and Simonsen (1994); Kensing and Winograd (1991); Simonsen (1994, 1996, 1997); and Simonsen and Kensing (1994, 1997).

2. WHY A METHOD, AND WHAT KIND OF METHOD?

In the years of outsourcing and BPR, many organizations have chosen to outsource costly and hard-to-manage software development. Bansler and Havn (1994) referred to this as "the 'industrialization' of information systems development" (p. 707), and they argued that in the future, most IT systems will be based on prefabricated generic systems.

In the same way that prefabricated walls, beams, and doors have not made architectural design irrelevant, we have found that the increased use of generic systems does not rule out a need for customized design. We argue that it is the job of design, based upon a thorough understanding of the organization in question, to investigate which generic systems are adequate, as well as how to reorganize work accordingly. Often generic systems need to be customized and supplemented with the design of organizationally specific systems to fulfill a coherent solution. It is these parts of systems development that we call design and that our method deals with: the analysis of needs and opportunities and the preliminary design of functionality and form. An organization may carry out a design project in cooperation with either internal IT specialists or external consultants. These we refer to as IT professionals, and they may or may not participate in the succeeding development and implementation activities.

In Figure 1, we combine Bansler and Havn's (1994) project model for industrial software development with our experiences. In this model, the organization relies on outside contractors for software development.

The organization's IT department (or external consultants) in cooperation with the user departments performs the design and specification of one or more coherent visions for change ("design") and then prepares a contractual bid ("Contractual bid and selection"). The chosen contractor then gets the contract of delivering generic IT products or developing organizational-specific systems ("Delivery and/or development of IT"). In parallel, the IT department performs "Delivery management". This involves quality control of deliverables from contractors. It also includes facilitating the organizational implementation by working with the user departments, external contractors, and other involved parties. There are major managerial decision points after "design" (e.g., which of the proposed solutions to go for) and as part of "Contractual bid and selection" (e.g., which contractor to choose). For IT professionals, this means taking on a role similar to that of an architect. Besides designing a building, the architect often is in charge of the overall supervision when the building is constructed. A particular instance of this

model occurs when the organization's IT department chooses also to bid on the contract, in which case, the systems development might take place as in-house development (Grudin, 1991).

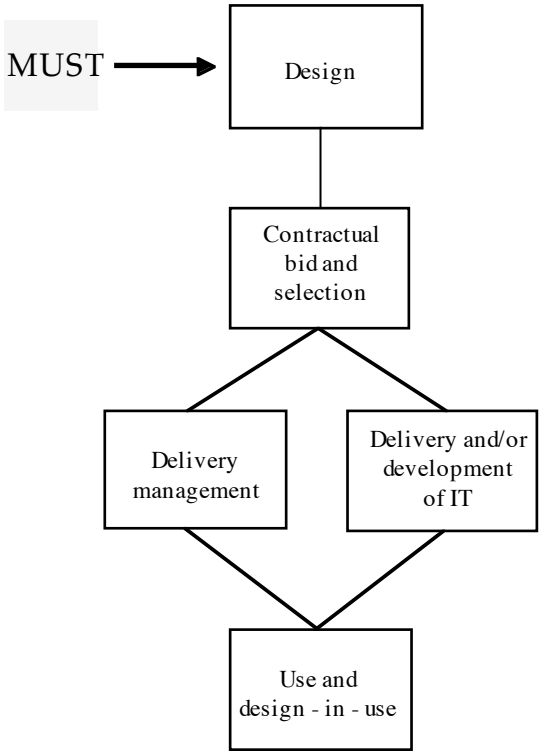

Figure 1. Project model for IT development.

The model (as well as Bansler & Havn, 1994) indicates that IT professionals need special skills to deal with people in the user organization on their own basis, and not solely on a technical basis. IT professionals have to handle complex and open problem situations. We see a method as a resource available for IT professionals facing such situations, rather than as a recipe to be followed step by step. IT professionals need methodological support for the activities that take place in an organization before a contractual bid and selection can take place and before the organization can decide which generic systems to select and purchase. We propose the MUST method as support for IT professionals responsible for these initial parts of IT development. We have learned that for IT professionals to integrate new work practices, a description of the method is needed, and the method also has to be

supplemented with other activities. We are currently involved in the supervision of ongoing projects involving IT professionals who are integrating the MUST method as part of their work practices.

The MUST method has been developed to support the design activity, as shown in Figure 1. The method is coherent in the sense that it deals with all activities within its application area: analysis of needs and possibilities, generation of visions for change, project management, and planning for technical and organizational implementation. Most Scandinavian PD researchers, coming from a background in trade union projects, have not explicitly dealt with activities related to management (see, however S. Bødker, 1996). We want to stress that for design ideas to be implemented, establishing and maintaining relations with management is crucial when designing in an organizational context.

Design in an organizational context is an open-ended process. The objective of the design project is to investigate the situation and provide information for a decision about how to proceed. If appropriate computer systems can be identified, the overall functionality and form of such systems are outlined. The results of a design project should include a conceptual design in terms of a written document, sketches, mock-ups, and prototypes. We consider an evaluation of consequences of implementing the design, as well as a plan for the implementation, to be parts of the result, too. Based on a design proposal, it should be possible for the organization to say "stop," to say "more design is needed," or to proceed in purchasing and developing the proposed design. The project may afterwards proceed to development and implementation, but we consider these parts of systems development to be outside the application area of our method.

We see organizations as frameworks for cooperation as well as for conflicts. Groups and individuals participating in design should be expected to have common as well as conflicting goals. The role of IT professionals is neither to cover up nor to solve political conflicts in design. Rather they should help the parties formulate their visions and leave it to them to solve conflicts in relevant fora (see Principle 2 in section 3.2). A good design most often is a mix of tradition and transcendence (Ehn, 1988). One reason for bringing in IT professionals is to transcend the tradition. At least one person in the organization has considered that some of the current ways of doing things have lost their rationale, or they have considered that new technological opportunities are worthwhile investigating. However, IT professionals need to understand and respect the existing traditions in an organization, both as a

way of maintaining - or establishing - credibility, and to understand the rationale behind phenomena that otherwise can be perceived as odd by an outsider.

We want to emphasize that an important ethical issue involved in applying our method - and for participation in general - is that if management wants job cuts or other drastic changes, this should be announced up front. Otherwise an important ethical principle will have been violated and, as a consequence, participation will be made more problematic and difficult in future projects. This does not imply that drastic changes cannot be realized by a participatory approach. We have experienced drastic changes in work organization as part of design projects, as well as job cuts just before the project started, but the users knew and accepted the objectives beforehand (Kensing et al., 1998).

3. THE SIX PRINCIPLES

Our method is grounded in six principles and offers a set of techniques and ways of representing current work and the envisioned computer-based systems. We consider the principles to be indispensable, although the techniques and representation tools may be chosen by the IT professionals according to their preferences and understanding of the situation in question. In this section we present each of the six principles and illustrate various techniques, representation tools, and principles of organization, when appropriate.

3.1. Principle 1: Participation

A large proportion of the software installed in organizations is never used. The primary reason for this is that IT professionals have not understood the specifics of the organization in question (Boehm, 1981; Bullen & Bennett, 1990; Lederer & Sethi, 1991; Lyytinen & Hirschheim, 1987; Orlikowski, 1992). Participation is a way of increasing the chances for a design to correspond with real needs and to be used as intended.

There have been both pragmatic and political arguments to participatory design (Ehn, 1988; Greenbaum, 1993). The pragmatic argument stresses that participation between IT professionals and users enables a mutual learning process and facilitates the development of an envisioned computer-based system. IT professionals need knowledge of the use context, users need knowledge of the technological options, and these should be developed in a colearning process. In our projects, this view has been acknowledged by users, management, and IT professionals. Further, in this article, we argue that

for a design vision to be realized, not only does an IT solution need to be technically correct, but the design team also needs to focus on anchoring the vision in the organization. This requires the design team to engage multiple participants in the design endeavor. It is the responsibility of the IT professionals to organize a participatory design process, and it is the responsibility of the management to provide users with the necessary time and information to participate in this process.

Political arguments to participatory design stress users' rights to influence their own working conditions and that this should be taken care of by their local union representatives (Ehn, 1988; Greenbaum, 1993; Kyng & Mathiassen, 1982; Nygaard, 1975). From the very start of the Scandinavian trade union projects, it has been a key issue to ensure that users get time off to participate and that trade unions should build up their own competence apart from the management-controlled systems development process (Ehn & Sandberg, 1979; Kyng & Mathiassen, 1982; Nygaard, 1975). We have great respect for these projects. However, we realize that IT professionals need to be pragmatic, too. In the years of downsizing, with the decrease of unions' power combined with the increase of employees striving to build a career - or hold on to their jobs - we have experienced very valuable user representatives. In spite of the fact that users were not given time off for participation and trade union participation has been low, or non-existent, the users were most eager to participate in project groups and as informants**.**

3.2. Principle 2: Close Links to Project Management

Project management deals with the division of labor in the project, how the project is designed as a process, quality control, and how conflicts are dealt with. We deliberately include establishing close interaction between project management and activities related to the design proper as a principle, because it has not been dealt with explicitly in PD literature.

We recommend a division of labor between a design team and a steering committee. The design team should consist of a combination of IT professionals and future users. They are the ones responsible for carrying out the project and for informing management and all future users. The steering committee should include managers of the involved organizational unit(s); the manager of the IT department, if any; and one or two user representatives[2].

---

[2] In some organizations the local union (by law or agreement) has a say in relation to

The design team must decide how it will organize the process of developing an understanding of the organization's needs and possibilities, developing visions of future computer-based systems, and sketching plans for technical and organizational implementation. Designing the project as a process is of special concern in dealing with the early design activities, because they are characterized more as problem setting than problem solving (Lanzara, 1983; Schön, 1983). We do acknowledge that for a group of IT professionals to be efficient, they need to rely on a set of standard techniques, representation tools, and ways of conducting projects. However, each project needs to be designed according to an understanding of the specifics of the actual context.

As described in further details in Section 4, we suggest the project be designed around the following five main activities: (a) project establishment, (b) strategic analysis, (c) in-depth analysis of selected work domains, (d) developing one or more visions of the overall change, and (e) anchoring the visions. Each activity produces knowledge that allows the design team to inform all future users and allows the steering committee to focus on the type of decisions that the design team needs to make in order to proceed. This enables the steering committee to make decisions on a qualified basis, thus minimizing risks in the ongoing interpretations of the project's goals, and of developing unrealistic visions.

Design is also a political process where groups and individuals have common as well as conflicting goals (Andersen et al., 1990). The steering committee is responsible for supervising the design project, dealing with potential and manifest conflicts, and making decisions based on information provided by the design team. We suggest leaving it to the steering committee to deal with the conflicts generated or becoming manifest in relation to the project. It is not up to the design team to solve the political controversies, but the team does have a role in providing a sound basis for dealing with them and in seeing that they are dealt with in the relevant fora. This has been emphasized in most of our projects (see K. Bødker, 1990; K. Bødker & Kensing, 1994; Kensing et al., 1998; Simonsen, 1994; Simonsen & Kensing, 1994; 1997).

We suggest three techniques for the design and the continuous evaluation of both the process and the product: project establishment, planning with baselines (Andersen et al., 1990), and reviews (Freedman & Weinberg, 1982).

---

development of new IT systems. If this is the case, we recommend that shop stewards become members of the steering committee. If this is not the case, users should be given the opportunity to appoint representatives.

3.3. Principle 3: Design as a Communication Process

In earlier work (Kensing & Munk-Madsen, 1993), we created a model of the communication between users and IT professionals (see Figure 2). The model is based on two distinctions: dealing with three domains of discourse and two levels of knowledge. The model is *not* a process model. It depicts six areas of knowledge that need attention. How knowledge in these areas is developed is dealt with in Section 4.

	Users' present work	New system	Technological options
Abstract knowledge	Relevant structures of users' present work	Visions and design proposals	Overview of technological options
Concrete experience	Concrete experience with users' present work	Concrete experience with the new system	Concrete experience with technological options

Figure 2. Six areas of knowledge in user-IT professional communication.

"Users' present work" includes work practice, organization of work, use of IT, products and services, relations to customers, clients, and suppliers, history of recent major changes, management strategies and style, and so forth. "New system" includes envisioned technology in relation to new work organization for the specific work domain. "Technological options" incorporates general knowledge and experiences with IT and its relation to work organization. The three domains reflect both the users' and the IT professionals' typical prerequisites in terms of knowledge and understanding prior to entering the design process. At the outset, the users have knowledge of their present work and of organizational options. The IT professionals have knowledge of technological options with regard to hardware and software. At the outset, this is all we can expect them to know.

The second distinction between abstract knowledge and concrete experience expresses that we need to deal explicitly with two levels of knowledge. Just as prototyping is a powerful approach for developing concrete experience with visions and design proposals, we argue that IT professionals need concrete experience with users' present work practices to understand and evaluate the relevance of oral or written descriptions of these practices. As we argue in the next section (Principle 4), it is by iterating between these two levels of understanding that the design team is able to develop the needed insight.

It is the responsibility of IT professionals to choose the techniques and representation tools that allow them to establish a communicative process with users, through which they are able to jointly develop knowledge within these six areas. Section 4 provides examples of techniques and representation tools for this purpose.

3.4. Principle 4: Combining Ethnography and Intervention

We apply a combination of ethnographic techniques and intervention in an iterative approach to design. We strive to select carefully the area and the mode of intervention, based upon what we have learned by applying ethnographic techniques - in contrast to BPR (Hammer & Champy, 1993, p. 207). Ethnography and intervention contrast in terms of their basic approaches and intended results: ethnographers originally strove not to change the phenomena they were studying, whereas interventionists deliberately set up activities to change the organization, to learn from the reactions to the change. However, we have experienced that at a practical level, combining the two approaches and iterating between them has been an effective way to learn about the organization and also an important resource in generating realistic visions of future use of technology (see K. Bødker & Kensing, 1994; Kensing et al., 1998; Simonsen, 1994; Simonsen & Kensing, 1994, 1997).

Ethnography: Firsthand Encounters

Blomberg et al. (1993) stated that "to learn about a world you don't understand you must encounter it first hand" (p. 125). It is crucial for IT professionals to develop a thorough understanding of users' present work for the design to reflect - in a realistic way - the norms and traditions of the organization. A design should be realistic in the sense that it reflects an appreciation of the rationale given by members of the organization and in the sense that the organization is geared to meet the challenges of the envisioned design. Through detailed studies of the organization's present situation, we try to "measure" the organization's needs and readiness for change (Christensen & Molin, 1983). We try to avoid an extreme futuristic design or a design of which the greater portion will never be used. We have found ethnographic techniques helpful in accomplishing this (K. Bødker & Kensing, 1994; Kensing et al., 1998; Simonsen, 1994; Simonsen & Kensing, 1994, 1997).

Blomberg et al. (1993) suggest descriptions in terms relevant to those being studied, in contrast to applying traditional IS techniques and their formalisms.

The latter, when at their best, suggest interviews with future users but are relying on the IT professionals' predefined conceptual frameworks. In Kensing and Munk-Madsen (1993), we argued that by going back and forth between observing users' work practice and producing descriptions (or interpretations, if you like) of these practices, IT professionals and users are able to develop an understanding of the current practices that are relevant in design.

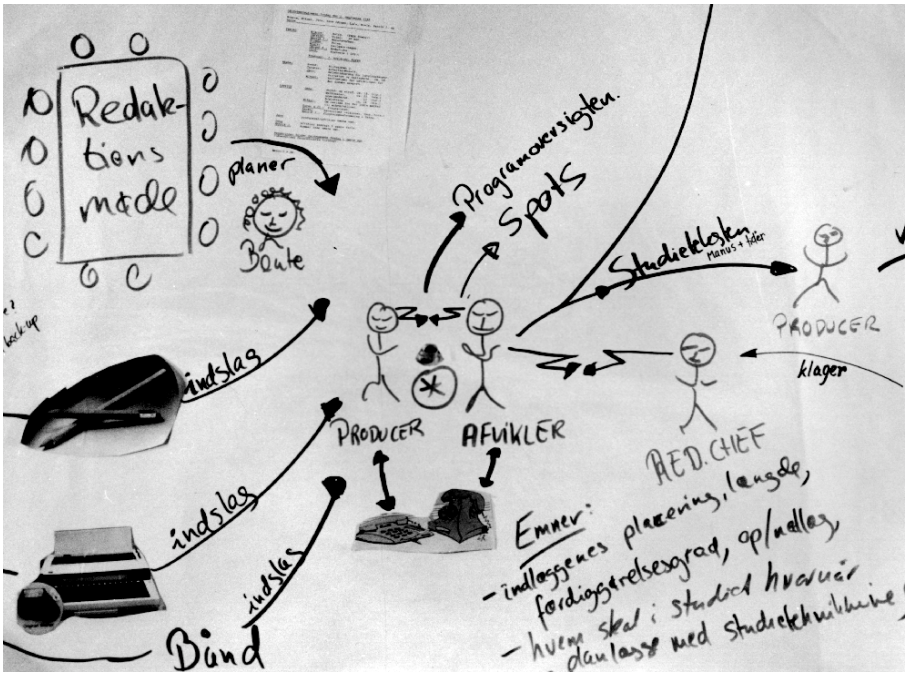

Figure 3. Excerpt from the collages (text is in Danish)

Formalisms play a minor role in the MUST method. In later parts of the development process, they are powerful tools, but when working with users without a technical background, we can easily do without them. We suggest using plain text, freehand drawings, sketches on large sheets of paper (e.g., representing communicative structures, the relation between work organization and the use of current/envisioned technology, and so forth). The closest we get to using formalism with users is when we model information flows and data structures for the purpose of prototyping.

We recommend two types of descriptions (reflecting the abstract/concrete distinction in Figure 2). One description is stated in language based on users' categories and presented in plain text or visualized - for example, by collages (see Figure 3, from K. Bødker & Kensing, 1994) or by wall graphs

(Simonsen, 1994; Simonsen & Kensing, 1994, 1997). The other description points out current domains and creates envisioned structured domains that might benefit from new IT systems (Dahlbom & Mathiassen, 1993; Winograd & Flores, 1986). For these, we use problem lists (Kensing et al., 1998) or maps (Lanzara & Mathiassen, 1984). The first type we have found useful in detecting and evaluating the relevance of the latter, which in turn is needed for further design purposes.

Intervention

Interventionists deliberately set up activities designed to change the organization. As Dahlbom and Mathiassen (1993) put it, "only by trying to change it [the organization] will we come to really understand it" (p. 169). The presumption is that through creating a change, key factors of the organization and its members' perception of it become observable - factors that might not be mentioned, for example, in interviews.

Schön (1983, 1992) described design as a reflective conversation with the materials of a design situation. According to Schön, intervention happens in the mind of the designer or through conversations among designers, rather than in the physical world, as with, for example, prototyping or organizational experiments. This type of intervention is less expensive in terms of time and potential consequences and, thus, preferable, but sometimes imagination is not enough and "real" experiments need to be carried out.

Iterations

Two types of iteration interplay when we combine ethnographic techniques with intervention. First, iterations between interviews and observations allow IT professionals to be aware of the discrepancies between what people say they do or want to be able to do and what IT professionals as outsiders are able to observe them doing. In other words, iterating between interviews and observations helps IT professionals handle the say/do problem (Blomberg et al., 1993; Gougen & Linde, 1993). Second, iteration between using ethnographic techniques and intervention may be used to confront users with these discrepancies between what is said and what is done. In K. Bødker and Kensing (1994), we used the detection of such discrepancies as the input for a design workshop[3]. Others suggest the use of rapid prototyping for similar

---

[3] Beforehand we had formulated provocative statements highlighting the differences between what users told us and what we were able to observe. These statements dealt with their current practice as well as with the relation between these and their ideas for IT

purposes, such as Mogensen (1992) who suggests the term *provotyping*. Yet others, such as Blomberg et al. (1996), suggest case-based prototypes. We suggest the two types of iteration even prior to prototyping.

3.5. Principle 5: Co-development of IT, Work Organization, and Users' Qualifications

IT is introduced because someone - usually management - wants change. However, projects far too often focus solely on IT systems, leaving it to the users to struggle with the organizational implementation afterwards and reducing educational aspects to training the functionality of the systems.

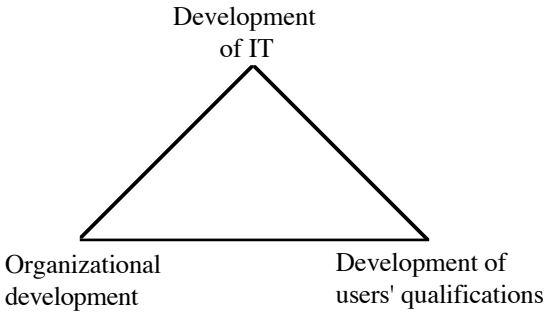

Figure 4. Co-development in related domains.

Since the early 1970s, Mumford and associates have worked on a sociotechnical approach (see Mumford, 1972, 1993; Mumford, Land, & Hawgood, 1978) advocating development of the social and technical systems more or less parallel to each other. The approach was heavily critiqued by Scandinavian researchers involved in trade union projects in the mid- to late 1970s. The critique was two-fold. From an ideological point of view, the approach to users' participation and control was evaluated as too narrow; from a technical point of view, the proposed techniques were evaluated as naive and as not addressing relevant aspects (Ehn & Sandberg, 1979; Kyng & Mathiassen, 1982). However, we owe to the sociotechnical approach the double focus on organizational and technical issues and for including management in a participatory approach. The sociotechnical approach even included prototyping as early as 1978 (Mumford et al., 1978).

---

support. This led to an evaluation of consequences of the design ideas and a clarification of which work practices they preferred, which in turn resulted in a modified set of requirements.

We recommend, as indicated in Figure 4, including a third issue in this co-development process - users' qualifications because we have seen too many systems that are only partly used because users have never been properly introduced to them. "It seems that the money ran out," as one user stated it. Educational activities help users (re-)gain control over their jobs and allows them to be more efficient.

Users need the qualifications to operate the systems that are supposed to support their work. However, this is often neglected in practice, and training is often organized around the functionality of the system rather than around the users' daily work. A design team's report should suggest who should receive education, how much education (in terms of content and time), how the education will be organized, and an estimate of the costs. If a new division of work is part of the design, or if new products or services are part of the overall vision of change, we see it as part of the design report to suggest adequate educational and training activities and an evaluation of the costs. Finally, we suggest an initial and ongoing introduction of user representatives to the method used in the project, as well as to what is expected from them in terms of involvement, relation to colleagues, and the specific tasks they will be participating in. The reason that we stress the importance of informing users of these issues is not in any way to hinder them from suggesting or taking other initiatives themselves. The point is that far too often we have seen projects where users were unaware of the scope and rationale of their participation.

All in all, a design project needs to address, plan for, and estimate the costs of taking care of technical, organizational, and educational issues. This should be done to produce a sustainable basis for the organization's decision making and for the succeeding development of the technical and organizational implementation to constitute a coherent whole.

3.6. Principle 6: Sustainability

The early design activities are a first step in introducing sustainable IT. We deliberately use this ecological concept as a metaphor in an attempt to capture an overall picture of the use of the method. In ecology, the concept of sustainability refers to a balance between the utilization and the protection of the earth's resources in order not to destroy the basis of mankind. There is a growing awareness of problems, alternative products and production processes are being developed, and the market is slowly adapting. We see a modest start of a similar process in the development and use of IT systems.

Negative consequences have been seen. Some IT systems have been designed or introduced in ways that made it difficult for users to use/develop their skills and experience as part of their job (see, e.g., Sachs, 1995). Often IT systems have failed economically, too - expected rationalization did not materialize, and projects ran far over budget. In such projects, scarce resources such as money and users' qualifications were not taken properly care of.

Researchers and practitioners have developed alternatives - regarding processes within PD and also regarding products within computer supported cooperative work - thus providing a basis for using and developing valuable resources in organizations. Users and managers have shown an interest in alternative products and processes. Of course they might not always agree on the positive and negative consequences of the application, but we have seen a willingness to have such issues dealt with up front in design projects (Kensing et al., 1998). What still remains is for IT professionals - on a larger scale - to be introduced to a coherent method for participatory design. This is the ambitious goal of the MUST method.

What is needed is a change of attitude for most managers and IT professionals. They need to experience through practice the effects of leaving the traditional expert strategy, the result of which sometimes has been completely reversed. The way many systems work shows that rationality has lapsed into irrationality. Such cases are often reported in the news and have been documented by a wide range of ethnographic studies. However, in working with managers and IT professionals in most of the 10 projects informing our method development, we have experienced an increasing awareness of the pitfalls in the predominant practice, as well as a willingness to experiment with alternatives.

The pragmatic argument for participation is related to the principle of sustainability. The MUST method suggests a high degree of user participation in order that new IT systems fit with preferred work practices, and the method supports the organization in an up-front uncovering and dealing with conflicts arising in relation to the introduction of IT (see Principle 2). Users, managers, and IT professionals in our projects sometimes found this cumbersome, but compared with previous experiences, they found it helpful in laying the basis for the proposed change.

4. FIVE MAIN ACTIVITIES CONSTITUTING THE DESIGN PROCESS

In the MUST method, the overall design process is constituted by five main activities: (a) project establishment, (b) strategic analysis, (c) in-depth analysis of selected work domains, (d) developing visions of the overall change, and (e) anchoring the visions. The main activities, each having their own purpose, support a stepwise decision-making process. Iterations are recommended, especially between the first and second activity and between the third and fourth activity. The fifth activity should be seen as an ongoing concern throughout the project.

I Section 4.1 through Section 4.5, we discuss each of the main activities, and in Figures 7 through 11 we give examples of techniques, representation tools, and principles of organization taken from a recent project. To give the reader an idea of the complexity of the organization and the suggested overall design, we refer to Figures 5 and 6.

The examples focus on the procedural aspects of an application of the method. We refer the reader to Kensing et al., (1998) for details on the intermediary and final products that the design team produced.

The purpose of the project was to bring multimedia support to P3, a music channel of the Danish Broadcasting Corporation, DBC. To help the reader contextualize the example, we present some of the background and the overall design of computer support that resulted from the project (see Figure 6).

DBC is the only Danish public station which is entirely funded by citizens paying a license fee. The corporation is divided into a TV section and a radio section. DBC has undergone several changes since its monopoly was broken in the late 1980's. While the former CEO had publicly announced that he didn't see it as his job to contribute to the growth in unemployment, the new CEO announced major restructuring for the purpose of producing more TV and radio for less money. These changes include: A shift in the corporation's profile from a production company to a broadcasting company; new IT support for administrative and managerial purposes; a different division of labor among journalists, technicians, and administrative staff; a shift from analogue to digital technology; restructuring of the IT departments; and considerable layoffs.

The Radio section comprises 3 national radio channels, a news group, and 9 regional radio stations. There is competition among the national channels and the regional stations, as well as between these and Danish and international radio and TV stations. The Radio strives to meet the competition by the introduction of new technology, by reducing costs, and by sharpening the profile of the programs.

Figure 5 (part one). Introduction to the design context (from which later examples are drawn).

P3 has 140 employees broadcasting 24 hours of radio daily all year round. When the project started the channel had very little computer support. It had only on line access to news agencies (NewsStar) and access to mainframe systems, like library systems used for searching in a database for the corporations music titles and earlier broadcasted programs. As to production and broadcasting, the equipment was analogue, except for experiments with hard disk editing of pre-produced features and computer based selection of music for broadcasting during the nights. In addition, typewriters, photocopiers, and text processors were used for the production and distribution of plans, manuscripts and management memos. The focus of the project was on the production of programs, while administrative and managerial aspects were only to be considered to the extent that production activities would generate relevant information for these purposes.

In relation to the P3 project, the most important parts of DBC's IT strategy are:

- that the technological platforms for office work and for the production and broadcasting of TV and radio will merge,
- that BPR is needed in relation to many work processes and that selective outsourcing should be applied for example in relation to development of applications, networks, installation of PC's, support, and the maintenance of installations,
- that new information systems should be based on standard systems, client/server technology, and a new wide band local area network (Intranet),
- that IT and data communication will be the strategic tools for reaching the right customers with the right products at the right time, as well as for reengineering internal work processes.
- that the key factor to success is that management and employees in the radio will be actively engaged in the control and implementation of IT.

As depicted in Figure 6 the project developed a coherent vision for changing the ways in which programs are produced. *MS-office* is used for writing stories (Word) and for communication purposes (E-mail). *Host* allows access to DBC's administrative systems. Journalists have access to *News Agencies*, the *Internet/WWW*, and an *Event Calendar* for research purpose. They search in *Sound Databases* for music titles and earlier broadcasted material. *List of Ideas* holds the potential stories for a given program. While working on a specific story a journalist uses *Program Element* to type in the information needed during broadcasting. *Digital Recording and Editing* is used for the production of each element in a program which is then stored in the *Pool* from where the producer drags and drops them into the *Manus* (his manuscript). The required reports about broadcasted programs are generated automatically by the *Report Generator*. *Video Links* support the communication between the studios and the offices in which the journalists work on their stories before they go to the studios.

Figure 5 (part two). Introduction to the design context (from which later examples are drawn).

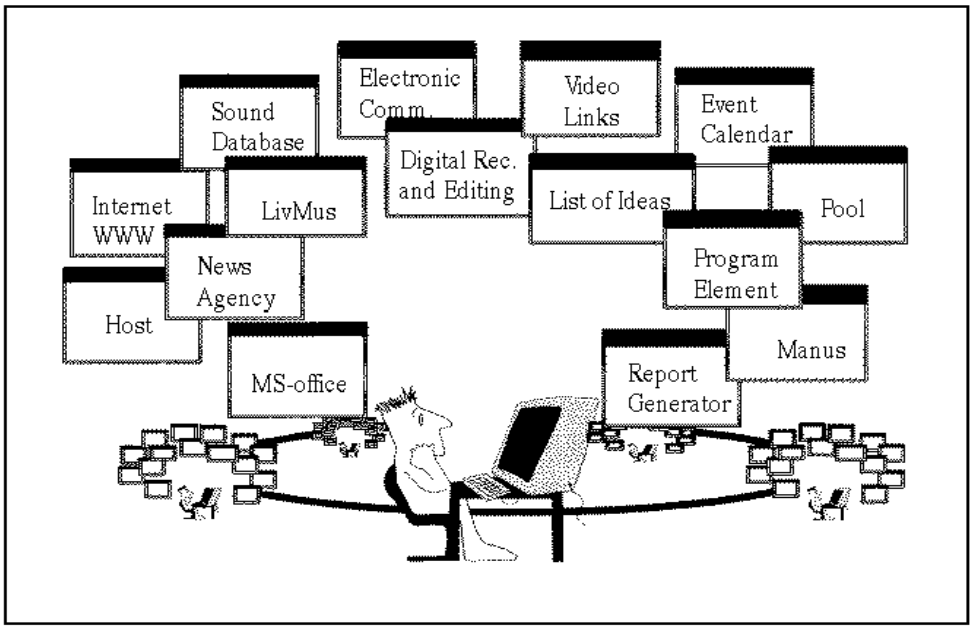

Figure 6. The suite of systems included in the design team's vision for change.

4.1. Project Establishment

We always recommend starting with project establishment (Andersen et al., 1990). This is a systematic technique supporting the clarification and negotiation of the aim, level of ambition, scope, and conditions of the project. The technique also suggests activities for the design team in deciding which tools and techniques it will use to conduct the project, as well as for establishing the team as a social unit. Although many projects start out from a rather loose description, project establishment provides the steering committee and the design team with a sound basis for the succeeding project activities.

In Lanzara's (1983) terms, project establishment is a reframing process. We have often experienced that management and users have had rather specific ideas of which IT systems are needed, but the problematic situation leading to the solutions had not been analyzed properly (see Simonsen, 1994, 1996, 1997; Simonsen & Kensing, 1994, 1997). We find that it is the responsibility of IT professionals to question such ideas, and project establishment is the first attempt in that direction.

Project establishment involves

- An initial analysis to understand the purpose of the project. This includes presentation rounds in various organizational units, interviews, observations, and an initial analysis of materials and artifacts used.
- An identification of critical success factors - what the project needs to fulfill.
- Meetings to negotiate the conditions of the project.
- A hearing4 of all involved actors on the basis of the final (or draft) project charter.
- Project planning and writing or negotiating the project charter, which is the basis for the steering committee's and the design team's decision about (and commitment to) how to approach the project.

Figure 7 presents an example of project establishment.

The design team started with only IT professionals, so part of our concern was to have the organization find user representatives. This was taken care of by the deputy manager who appointed two journalists and a secretary, while the local unions did not intervene. A steering committee was formed consisting of the deputy manager (chairman), three middle managers from the radio channel, and the IT manager. The project team interviewed the IT manager about IT strategies, and they interviewed the P3 managers about the purpose of the project and about how the project should be related to corporate and channel strategies. The project team read various corporate strategic documents as well. The IT professionals observed an editorial unit for a full day and interviewed a producer, a couple of journalists, a technician, and a secretary. The design team had a one day meeting where the project was outlined and a preliminary plan was drawn up. Two of the IT professionals made a draft of the project charter, which was then approved by the full design team with minor changes. The team then negotiated the draft with the steering committee. The design team and the steering committee agreed upon the aim, level of ambition, scope, and conditions of the project. The final project charter laid down the technological platform (Windows NT for clients and servers) and a package of standard office systems (MS Office). The only major point of disagreement was that the deputy manager didn't think the three users in the design team should spend as much time on the project as we had suggested. We settled this disagreement by agreeing that the three users spend what corresponds to one full time person, while the IT professionals should spend an equivalent of two full time persons. This first part of the project was in progress over 6 weeks (calendar time), and the design team spent a total of some 3 person weeks.

Figure 7. Example of project establishment.

---

4 By a hearing we mean that the involved actors are informed about the given subject matter with the possibility of commenting on it.

4.2. Strategic Analysis

The purpose of strategic analysis is to clarify and delimit which work domains should be in focus in the design project. This is often rather unclear, even if the organization has a business strategy and a related IT strategy (Kensing et al., 1998; Simonsen, 1994, 1996, 1997; Simonsen & Kensing, 1994, 1997). Like project establishment, this too is a reframing process.

In some cases, strategic analysis may be a part of project establishment. If the organization is unable to define the focus of the design project in an adequate way, or if there are conflicts as to which areas should be given priority, we suggest that the strategic analysis be handled separately (see Simonsen, 1994, 1996, 1997).

The manifest result depends on the degree to which the organization in question already has a business strategy and a related IT strategy, and on the degree to which the involved parts of the organization see the relation between these and the current project. Strategic analysis clarifies the potentials for investments in IT support and investigates organizational, economical, and technical limitations. It involves development of an understanding of the organization's situation in a competitive market, which parts of the organization need to be strengthened and how this relates to the current project, identification and analysis of customers and suppliers (internal or external), and which products and services the organization should provide. The focus of strategic analysis is on the functional requirements of the environment on the organizational units in question (Schmidt, 1988; Simonsen, 1994).

Strategic analysis is primarily a management related activity. For this activity, the MUST method suggests

Interviews of managers; the IT manager, if any; and representative users, customers, and suppliers, as well as observations of key activities.

- Document analysis of (possible) strategic plans, IT strategies, and market surveys.
- Functional analysis (Schmidt, 1988; Simonsen, 1994, 1997).
- A hearing of all involved actors organized by the steering committee. The purpose is to collect comments for an eventual modification of the strategic analysis and the project charter. Equally important, such a hearing ensures that all actors involved are informed about the objectives of the third activity: in-depth analysis of selected work domains.

Strategic analysis leads to a decision situation, whereby it is decided which work domains should be further analyzed and subsequently supported by IT. Figure 8 presents an example of strategic analysis.

In the P3 project the strategic analysis was carried out in parallel with the establishment of the project. In addition to what is reported above, we read the CEO's strategic plans, the Radio's IT strategy, and surveys of listener behavior. However, during the in-depth analysis of P3 (see Figure 9) we learned that another IT department at the Radio was running a similar project for the regional stations. This project had a different perspective and rationale than the P3 project. Since this had not been brought to the P3 design team's attention during the strategic analysis, we found it necessary to arrange meetings with the regional design team to learn more about them and their projects and to coordinate between the projects. During this process we realized that the two IT managers from each of their respective departments were competing with each other. The design teams decided to give up their attempt to coordinate. Seen though in retrospect, we should have insisted on coordinating.

The P3 design team also tried to coordinate with the department responsible for organizational development, since part of our project was to investigate a new division of labor among journalists, technicians, and administrative staff, which was included in the corporation's strategic plans. The organizational development department however declined our proposal to coordinate, partly because of its involvement in other projects, and partly because it wanted the management of P3 to state up front the goals of the reorganization. Instead P3 management and the design team saw it as part of the project to develop visions of technical and organizational changes in parallel.

Figure 8. Example of strategic analysis.

4.3. In-Depth Analysis of Selected Work Domains

The work domains pointed out by strategic analysis are in focus when in-depth analyses of current work practices are performed. The purpose of these in-depth analyses is to reveal and develop an understanding of the rationale behind current work practices ("users' present work" in Figure 2). The intention is not to map old practices into the new computer based system. However, we have experienced that users have good reasons for what they do, and that the rationale underlying current work practices is relevant for the design, even if management aims at rather drastic changes (Kensing et al., 1998).

The techniques proposed for developing an understanding of the work practice, of the use of current systems, and of the use of information are:

- Interviews and observations, where directly affected users at all levels are involved.
- Document analysis of documents used in the work practice.
- Thinking aloud experiments.

- Mapping (Lanzara & Mathiassen, 1984)
- Future workshops (Kensing, 1987; Kensing & Madsen, 1991)
- Workshops where the design team, perhaps supplemented by additional users, makes rich pictures (Checkland & Scholes, 1990), collages (K. Bødker & Kensing, 1994), or wall graphs of current work practices (Simonsen, 1994; Simonsen & Kensing, 1994, 1997).

IT professionals might need to make preparations for these activities and subsequently carry out, for example, modeling communicative structures (Kensing & Winograd, 1991) or cultural analysis (K. Bødker & Pedersen, 1991), which then should be reviewed by the design team and affected users.

Even though project establishment and strategic analysis have pointed out specific work domains, the analysis might lead to a conclusion that other domains need to be included in this activity, as well. In which case, the project charter is re-negotiated.

This main activity was the key focus of the design team during 10 weeks (30 person weeks). We carried out interviews, observations and thinking aloud experiments with one third of the employees of P3. Most of these were audio recorded, and rough transcripts were approved by each of the involved before they were condensed into accounts of the practices of the various communities that make up P3: editorial units consisting of a group of journalists, technicians, and administrative personnel; one-person editorial units; middle management; top management; and various support staff. In addition, the design team collected and analyzed work documents, and a number of the design team's meetings had the form of workshops. At the workshops, additional employees were invited to take part in mapping their individual work and its relation to their colleagues' work. For these sessions, we used free hand drawings on large sheets of paper that were attached to the walls. Some drawings focused on the communicative structure involved in radio production, others on temporal structures, yet others on information needed for and created in the process of planning, production, and administrative follow up of radio programs. The design team produced a list of problems and related suggestions for solutions. This enabled the steering committee to decide which areas should be supported and which design ideas should be developed further in the next main activity.

Figure 9. Example of in-depth analysis of selected work domains.

The results of in-depth analysis are descriptions of the current work organization; the use of IT; and the related problems, needs, and ideas for IT support. These descriptions are supplemented with an ordered comprehensive list of problems, needs, and related ideas for IT support and work organization. Other important results of this analysis are that users come to see their own work in the light of others' work, and IT professionals get to know users' concepts and categories, thus facilitating communication.

This leads to the third prototypical decision situation, where the steering committee decides which of the ideas for IT support should be given priority. Additionally we suggest a hearing of all involved actors and suggest that the design team collect the actors' comments for the purpose of an eventual modification. Figure 9 illustrates an in-depth analysis of selected work domains.

4.4. Developing Visions for the Overall Change

The development of one or more visions for the overall change is the central activity in the MUST method. We emphasize that the visions should not only deal with the functionality and the user interface of the suggested systems, but they should also include organizational change and changes in qualifications needed by the users (see Principle 5).

Ideas and visions are developed throughout the project, and they are often initiated in the very beginning of the project (Stolterman, 1991, 1992). They emerge in nearly all activities conducted in the project, but the purpose of this activity is especially to develop ideas and visions and form them into one or more coherent visions for change.

We suggest

- Visiting "similar" workplaces using new IT facilities.
- Holding future workshops (Kensing, 1987; Kensing & Madsen, 1991).
- Holding design workshops where the design team, perhaps supplemented by affected users, sketches on large sheets of paper the envisioned future work organization and its relation to new IT facilities.
- Sorting out design ideas - for example, by writing them on self-adhesive notes and grouping them on a wall.
- Carrying out data modeling.
- Making mock-ups and prototypes.

Again, IT professionals might have to make preparations for these activities and subsequently carry out, for example, information modeling and the development of prototypes.

The result of this activity is a design report, that states the aim of the project, sums up the analyses, and describes the suggested visions. The design report is supplemented with mock-ups and prototypes of the proposed IT systems.

The report holds an evaluation of positive and negative consequences of the suggested visions, regarding the organization as a whole, consisting of both organizational units and communities of users. For this task we suggest scenarios outlining how the work will be carried out when the visions are implemented (Clausen, 1993; Kensing et al., 1998; Kyng, 1995; Simonsen, 1994). Finally, the report includes estimated costs as well as a plan for purchase or development of IT systems, for technical and organizational implementation, and for the education and training of users.

The design team spent 10 weeks (30 person weeks) on developing coherent visions of computer support, organizational changes and related needs for new qualifications. The design team split up in two sub-teams each of which visited a radio station abroad with "state of the art" digital systems. We video-taped central work processes, took notes, and collected written material to inform each other. When the two sub-teams arrived back home, they held a number of design workshops - again using large sheets of paper - to map the relations between envisioned technology and work. At one workshop, we wrote all the needs coming out of the earlier activities, and all the design ideas on post-it notes. Grouping these on a wall revealed loose connections that had to be investigated further. For the purpose of prototyping, and for the subsequent programming, we developed data models on large sheets of paper, that were put up on the wall. In smaller groups, we then developed prototypes of the key subsystems. The prototypes were demonstrated for the whole design team and for the steering committee. The status of the prototypes did not allow for testing in real work situations. Finally, the team wrote a report that summed up the needs - they were related to the design ideas in a schema - and an estimate for their implementation was given. We developed a scenario of the envisioned new work practices, by giving an example of how an editorial unit would use the new technology to coordinate among unit members, with other editorial units, and with the editorial board. Each of the new systems were described in text and with illustrations. The consequences as to costs and as to a new division of labor were spelled out. The report also held a plan for the organizational implementation (for instance the employees were split up in ten groups for the training, and the equipment would be installed during the course) as well as for the technical implementation (for example in which order should the various subsystems be developed, which should be developed in-house, and which should be outsourced.) The design report was presented to the steering committee and at a hearing for all employees. The report was accepted, and the job of the design team came to an end. In the course of half a year, the proposed equipment and standard systems were purchased, the employees received the training, and the development of the organizational specific systems started.

Since only parts of the proposed design have been implemented, while a tender for the remaining parts has just been sent out, we feel that an evaluation of the successes and failures will have to wait until all the systems are up and running and the organizational implementation has taken place.

Figure 10. Example of developing visions for the overall change.

The design report forms the basis on which the steering committee decides which parts of the proposed design should be purchased as generic systems, some of which might have to be customized; which parts need to be developed especially for the organization; and which parts should be postponed or

perhaps rejected. Also, suggested organizational changes and training activities are decided by the committee. We suggest that the steering committee organizes a hearing of all involved actors, thereby collecting comments for the purpose of an eventual modification of the proposed design. Figure 10 presents an example of developing visions.

4.5. Anchoring the Visions

We use *anchoring* as a metaphor (Simonsen, 1994) that moves beyond the design/implementation dichotomy. For a vision to materialize, it needs to be deeply rooted in the organization. Its rationale needs to be understood by

- Management and the steering committee, who decide if it should be implemented.
- Those who will carry out the technical and organizational implementation - the latter including training activities.
- The users who will have to live with its consequences.

Because the mentioned actors are not all directly involved in developing the visions, time and resources must be set aside to make it possible for them to get to know the visions. Anchoring the visions is the job of the design team and the management of the involved parts of the organization. We see this activity to be orthogonal to the other four. It should be given attention in project establishment and in strategic analysis, and both the direct participation of users, as well as the suggested hearings, contribute to the anchoring activity. The purpose is to prepare for and even start the process of organizational change while still carrying out analysis and design activities. This guides why and by which means the design team and management interact with actors in the organization and maybe also outside contractors. In this respect, anchoring the visions is contributing to viewing design as a process of change.

A participatory approach to design is the central strategy in obtaining appropriate anchoring of the visions. This includes

- Meetings and workshops including developing, presenting and evaluating design ideas.
- Prototyping.
- Visits to other institutions using potentially relevant IT.
- Demonstration of IT products.

- Hearings.
- Scenarios describing envisioned future work practices supported by the proposed designs.

The design report and prototypes cannot convey everything that the design team learned throughout the project (Naur, 1985). Therefore appropriate anchoring requires that (part of) the design team has to cooperate, at least in an overlapping period of time, with those taking care of technical and organizational implementation. For IT professionals, this means having a role similar to that of architects. Besides designing a building, the architect is often in charge of the overall supervision when the building is being constructed. Figure 11 illustrates an example of anchoring visions.

Management wrote about the project in the departments' newsletters during the design period. The design team presented intermediate results to the management team and at the hearings. Since no journalists turned up for the first hearing, the project leader arranged to present and receive feedback on the work of the design team at regular meetings with each of the channel's editorial groups. During the interviews, observations, and thinking aloud experiments with one third of the employees, time was set aside for the design team to discuss more freely with the employees the rationale of the project and to listen to their ideas. The design team tried, but failed, to engage the organizational development department and those responsible for the training program. Finally, the external programmers that were to develop the organizational specific systems were selected. They read the design team's report, spent one day observing an editorial unit, and the prototypes were demonstrated for them. Part of the programming done by the external programmers took place at the IT department at the Radio. In this way, any questions that arose could be more easily handled.

Figure 11. Example of anchoring the visions.

5. SUMMARY

We have argued for the need of a separate design activity to produce a sustainable basis for further development and implementation of IT in an organizational context. Within the tradition of PD, we have presented a conceptual framework; a coherent method; and suggested techniques, representation tools, and principles of organization for this design activity. Figure 12 summarizes the MUST method's main activities and their corresponding decisions.

We have illustrated the MUST method as it was applied in one of the projects, through which the method has been evaluated and modified. Up until now the method has proven successful, even in design projects linked to job cuts and drastic changes in work organization. These projects have been carried out or supervised by the authors. The degree to which the MUST method, without

our direct involvement, is a useful resource for IT professionals in their work with users and managers is currently being tested in three organizations. We invite the reader in challenging the method.

Main activity	Leads to decisions about
Project establishment	Project charter
Strategic analysis	Work domains in focus
In-depth analysis of selected work domains	Problems, needs, and ideas for IT-support
Developing visions for the overall change	Coherent visions for change Evaluation of consequences Plans and estimates for implementation
Anchoring the visions	None (on going concern related to dissemination and feedback on project results)

Figure 12. The MUST method's main activities and corresponding decisions. The design team carries out all the main activities involving users and management as needed, hereby producing the basis for the steering committee's decisions.

NOTES

Background. This is a significantly revised and expanded version of a paper that was presented at the PDC'96-conference (Kensing, Simonsen, & Bødker, 1996).

Finn Kensing, ***Jesper Simonsen***, and ***Keld Bødker*** are associate professors in Computer Science at Roskilde University. They share an interest in the research areas of participatory design, computer supported cooperative work, and human computer interaction.

HCI Editorial Record. First manuscript received July 11, 1997. Revision received April 6, 1998. Accepted by Terry Winograd. - *Editor*

REFERENCES

Andersen, N. E., Kensing, F., Lundin, J., Mathiassen, L., Munk-Madsen, A., Rasbech, M., & Sørgaard, P. (1990). *Professional systems development: Experience, ideas and action*. Englewood Cliffs, NJ: Prentice-Hall.

Bansler, J. & Havn, E. (1994). Information systems development with generic systems. *Proceedings of the Second Conference on Information Systems,* 707-715. Breukelen, The Netherlands: Nijenrode University Press.

Blomberg, J., Giacomi, J., Mosher, A., & Swenton-Hall, P. (1993). Ethnographic field methods and their relation to design. In D. Schuler & A. Namioka (Eds.), *Participatory design: Principles and practices* (pp. 123-155). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Blomberg, J., Suchman, L., & Trigg, R. (1996). Reflections on a work-oriented design project. *Human-Computer Interaction, 11*, 237-265.

Boehm, B. (1981). *Software engineering economics*. Englewood Cliffs, NJ: Prentice-Hall.

Bullen, C. V., & Bennett, J. L. (1990). Learning from user experience with groupware. *Proceedings of the Conference on Computer-Supported Cooperative Work,* 291-302. New York: ACM.

Bødker, K. (1990). A cultural perspective on organizations applied to analysis and design of information systems. In G. Bjerknes, B. Dahlbom, L. Mathiassen, J. Stage, K. Thoresen, P. Vendelbo, & I. Aaen (Eds.), *Organizational competence in systems development. A Scandinavian contribution* (pp. 211-232). Lund, Sweden: Studentlitteratur.

Bødker, K. & Kensing, F. (1994). Design in an organizational context: an experiment. *Scandinavian Journal of Information Systems, 6*(1), 47-68.

Bødker, K., & Strandgaard Pedersen, J. (1991). Workplace cultures: Looking at artifacts, symbols, and practices. In J. Greenbaum, & M. Kyng (Eds.), *Design at work: Cooperative design of computer systems*. (pp. 121-136). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Bødker, S. (1996). Creating conditions for participation: Conflicts and resources in systems design. *Human-Computer Interaction, 11*, 215-236.

Carstensen, P. (1996). *Computer supported coordination*. Unpublished doctoral dissertation, Department of Computer Science, Roskilde University, Denmark.

Cash, J.I., McFarlan, F.W., & McKenney, J.L. (1992). *Corporate information systems management*. Homewood, IL: Business One Irwin.

Checkland, P., & Scholes, J. (1990). *Soft systems methodology in action.* Chichester, England: West Sussex.

Christensen, S., & Molin, J. (1983). *Organisationskulturer - kultur og myter, deltagerstyring, ledelse og omstilling, viden og erfaring, design,* [Organizational cultures - Culture and myths, participatory government, management and change-over, knowledge and experience, design]. Copenhagen, Denmark: Akademisk Forlag.

Clausen, H. (1993). Narratives as tools for the systems designer. *Design Studies*, *14*, 283-298.

Coad, P., & Yourdon, E. (1991). *Object-oriented analysis*. Englewood Cliffs, NJ: Yourdon Press.

Dahlbom, B., & Mathiassen, L. (1993). *Computers in context. The philosophy and practice of systems design.* Oxford, England: Blackwell.

Earl, M. J. (1993). Experiences in strategic information systems planning. *MIS Quarterly, 17*(1), 1-24.

Ehn, P. (1988). *Work-oriented design of computer artifacts*. Stockholm, Sweden: Arbetslivcentrum.

Ehn, P., Meggerle, T., Steen, O., & Svedemar, M. (1997). What kind of car is this sales support system? On styles, artifacts and quality-in-use. In M. Kyng & L. Mathiassen (Eds.), *Computers and design in context* (pp. 111-143). Cambridge, MA: MIT Press.

Ehn, P., & Sandberg, Å. (1979). *Företagsstyring och löntagermakt. Planering, datorer, organisation och fackligt utredningsarbete* [Management and employee power. Planning, computers, organization and union investigation]. Stockholm, Sweden: Prisma.

Freedman, D. P., & Weinberg, G. M. (1982). *Handbook of walkthroughs, inspections and technical reviews.* Boston, MA: Little, Brown.

Gougen, J.A., & Linde, C. (1993). Techniques for requirements elicitation. *Proceedings of the IEEE International Symposium on Requirements Engineering,* 152-164. Los Alamitos, CA: IEEE Press.

Greenbaum, J. (1993). PD: A personal statement. *Communications of the ACM*, *36*(4), 47.

Greenbaum, J., & Kyng, M. (Eds.) (1991). *Design at work: Cooperative design of computer systems*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Grudin, J. (1991). Interactive systems: Bridging the gaps between developers and users. *IEEE Computer*, *24*(4), 59-69.

Grønbæk, K., Kyng, M., & Mogensen, P. (1993). CSCW challenges: Cooperative design in engineering projects. *Communications of the ACM, 36*(4), 67-77.

Grønbæk, K., Kyng, M., & Mogensen, P. (1997). Toward a cooperative experimental system development approach. In M. Kyng & L. Mathiassen (Eds.), *Computers in context,* (pp. 201-238). Cambridge, MA: MIT Press.

Hammer, M., & Champy, J. (1993). *Reengineering the corporation: A manifesto for business revolution.* London: Nicholas Brealey.

Henderson, J. C., & Sifonis, J. G. (1988). The Value of strategic IS planning: Understanding consistency, validity, and IS markets. *MIS Quarterly*, *12*, 187-200.

Henderson, J. C., & Venkatraman, N. (1992). Strategic alignment: A model for organizational transformation through information technology. In T. A. Kochan & M. Useem (Eds.), *Transforming Organizations* (pp. 97-117). New York: Oxford.

Hughes, J. A., Randall, D., & Shapiro, D. (1992). Faltering from ethnography to design. *Proceedings of the Conference on Computer Supported Cooperative Work,* 115-122. New York: ACM.

Hughes, J. A., Randall, D., & Shapiro, D. (1993). From ethnographic record to system design: Some experiences from the field. *Computer Supported Cooperative Work (CSCW): An International Journal, 1*(3), 123-141.

Keen, P. G. W. (1991). *Shaping the future: Business design through information technology*. Boston: Harvard Business School Press.

Kensing, F. (1987). Generation of visions in systems development: A supplement to the toolbox. In P. Docherty, K. Fuchs-Kittowski, P. Kolm, & L. Mathiassen (Eds.), *Systems design for human development and productivity: Participation and beyond*, (pp. 285-301). The Netherlands: Elsevier.

Kensing, F., Bødker, K., & Simonsen, J. (1994). *An emerging approach to systems design: Experience from the MUST program* (Datalogiske skrifter No. 94). Roskilde, Denmark: Roskilde University.

Kensing, F., & Madsen, K.H. (1991). Generating visions: Future workshops and metaphorical design. In J. Greenbaum & M. Kyng (Eds.), *Design at work: Cooperative design of computer systems,* (pp. 155-168). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Kensing, F., & Munk-Madsen, A. (1993). Participatory design: Structure in the toolbox. *Communications of the ACM 36*(4), 78-85.

Kensing, F., Simonsen, J., & Bødker, K. (1996). MUST – A method for participatory design. *Proceedings of the Fourth Biennal Conference on Participatory Design,* 129-140. Palo Alto, CA: Computer Professionals for Social Responsibility

Kensing, F., Simonsen, J., & Bødker, K. (1998). Participatory design at a radio station. *CSCW: The Journal of Collaborative Computing,7*(3-4), 243-271.

Kensing, F., & Winograd, T. (1991). Operationalizing the language/action approach to design of computer-support for cooperative work. In R. K. Stamper, P. Kerola, R. Lee, & K. Lyytinen (Eds.), *Collaborative work, social communications and information systems*, (pp. 311-331). Amsterdam, The Netherlands: North-Holland.

Kyng, M. (1995). Making representations work. *Communications of the ACM*, *38*(9), 46-55.

Kyng, M., & Mathiassen, L. (1982*). Systems development and trade union activities: Information society for richer, for poorer*, (pp. 247-260). Amsterdam, The Netherlands: North-Holland.

Lanzara, G. F. (1983). The design process: Frames, metaphors, and games. In K. Briefs, C. Ciborra, & L. Sneider (Eds.), *System design for, with, and by the users,* (pp. 29-40). Amsterdam, The Netherlands: North-Holland.

Lanzara, G. F., & Mathiassen, L. (1984) *Mapping situations within a system development project: An intervention perspective on organizational change* (MARS-report No. 6, DAIMI PB-179). Aarhus, Denmark: Computer Science Department, Aarhus University.

Lederer, A. L., & Sethi, V. (1991). Critical dimensions of strategic information systems planning. *Decision Sciences, 22*, 104-119.

Lyytinen, K., & Hirschheim, R. (1987). Information systems failures: A survey and classification of the empirical literature. *Oxford Surveys in Information Technology*, *4*, 257-309.

Mathiassen, L. (1981). *Systemudvikling og systemudviklingsmetode*, [System development and system development method] (DUE-Report No. 5, DAIMI PB-136). Aarhus, Denmark: Datalogisk Afdeling, Matematisk Institut

Mogensen, P. H. (1992). Towards a provotyping approach in systems development. *Scandinavian Journal of Information Systems, 4*, 31-53.

Mumford, E. (1972). Job satisfaction: A method of analysis. *Personnel Review*, *1*(3).

Mumford, E. (1993). The ETHICS approach. *Communications of the ACM*, *36*(4), 82.

Mumford, E., Land, F., & Hawgood, J. (1978). A participative approach to the design of computer systems. *Impact of Science on Sociology*, *28*(3), 235-253.

Naur, P. (1985). Programming as theory building. *Microprocessing and Microprogramming, 15*, 253-261.

Nygaard, K. (1975). Kunnskapsstrategi for fagbevegelsen [Knowledge strategy for the trade union]. *Tidsskrift för Universitets- och Forskningspolitik. Nordisk Forum 6*, *10*(2), 15-27.

Orlikowski, W. J. (1992). Learning from notes: Organizational issues in groupware implementation. *Proceedings of the Conference on Computer-Supported Cooperative Work,* 362-369. New York: ACM.

Plowman, L., Rogers, Y., & Ramage, M. (1995). What are workplace studies for? *Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work*, 309-324. Dordrecht, The Netherlands: Kluwer.

Premkumar, G., & King, W. R. (1994). Organizational characteristics and information system planning: An empirical study. *Information System Research*, *5*(2), 75-109.

Rasmussen, J., Mark Pejtersen, A., & Goodstein, L. P. (1994). *Cognitive systems engineering*. New York: Wiley.

Sachs, P. (1995). Transforming work: Collaboration, learning and design. *Communications of the ACM*, *38*(9), 36-44.

Schmidt, K. (1988). Function analysis instrument. In G. Schäfer & R. Hirschheim (Eds.), *Functional analysis of office requirements: A multiperspective approach*, (pp. 261-289). Chichester, England: Wiley.

Schmidt, K. (1990). *Analysis of cooperative work. A conceptual framework* (Risø-M-2890). Roskilde, Denmark: Risø National Laboratory.

Schön, D. A. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books.

Schön, D. A. (1992). Designing as reflective conversation with the materials of a design situation. *Knowledge-Based Systems*, *5*(1), 3-14.

Shapiro, D. (1994). The limits of ethnography: Combining social sciences for CSCW. *Proceedings of the Conference on Computer Supported Cooperative Work,* 417-428. New York: ACM.

Simonsen, J. (1994). *Designing systems in an organizational context: An explorative study of theoretical, methodological, and organizational issues from action research in three design projects* (Datalogiske skrifter No. 52). Unpublished doctoral dissertation, Roskilde University, Roskilde, Denmark..

Simonsen, J. (1996). Involving customer relations in contextual design: A Case Study. *Proceedings of the 4th European Conference on Information Systems,* 1153-1161. Lisbon, Portugal.

Simonsen, J. (1997). Linking design to business strategy through functional analysis. *Proceedings of the 5th European Conference on Information Systems,* 1314-1327. Cork, Ireland: Cork Publishing Limited.

Simonsen, J., & Kensing, F. (1994). Take users seriously, but take a deeper look: Organizational and technical effects from designing with an ethnographically inspired approach. *Proceedings of the Third Biennial Conference on Participatory Design,* 47-58. Palo Alto, CA: Computer Professionals for Social Responsibility.

Simonsen, J., & Kensing, F. (1997). Using ethnography in contextual design. *Communication of the ACM*, *40*(7), 82-88.

Simonsen, J., & Kensing, F. (1998). Make room for ethnography in design! *The Journal of Computer Documentation*, *22*(1), 20-30.

Stolterman, E. (1991). *Designarbetes dolda Rationalitet. En studie av metodik och praktik inom systemutvekling*. [The hidden rationale of design work -

A study in the methodology and practice of system development] (No. 14.91). Unpublished doctoral dissertation, Institutionene for Informationsbehandling, Administrativ databehandling, Umeå Universitet.

Stolterman, E. (1992). How systems designers think about design and methods: Some reflections based on an interview study. *Scandinavian Journal of Information Systems, 4*, 137-150.

Suchman, L. (1995). Making work visible. *Communications of the ACM, 38*(9), 56-64.

Winograd, T., & Flores, F. (1986). *Understanding computers and cognition, A new foundation for design*. Reading, MA: Addison-Wesley.

Yetton, P. W., Craig, J. F., & Johnston, K. D. (1995). Fit, simplicity and risk: Multiple paths to strategic IT change. *Proceedings of the Sixteenth International Conference on Information Systems (ICIS'95),* 1-11. The Netherlands.

Yourdon, E. (1989). *Modern structured analysis*. Englewood Cliffs, NJ: Prentice Hall.