PD in the Wild;
Evolving Practices of Design in Use

Yvonne Dittrich[I], Sara Eriksén[II] and Christina Hansson[I]
[I]Department of Software Engineering and Computer Science,
[II]Department of Human Work Science and Media Technology,
Blekinge Institute of Technology
Box 520, SE-372 25 Ronneby, Sweden
+46 457 38 58 42, +46 457 38 55 65, +46 457 38 58 62
yvonne.dittrich@bth.se, sara.eriksen@bth.se, christina.hansson@bth.se

ABSTRACT

The when and where of participatory design has traditionally been set, primarily, by the software design project. However, modern IT networks with a variety of applications from different software providers, new web-design tools, and the integration of customization processes with ongoing version management, are just a few of the developments that are moving participation around IT design issues beyond the traditional software project. Using examples from a research project focusing on existing work practices and IT in use in public service administration, we explore various understandings of design, which challenge some of the assumptions underlying the basic framework of participatory design.

If design is seen as continually on-going, and intricately interwoven with use, this raises several important issues for participatory design. It highlights design for change. It points towards the need for reconsidering software design processes. It brings into focus issues of coordination between use, design in use and adaptation and development. Crucially, it raises issues about shop floor IT management, that is, organizational and technical support for local adapting, and continual design and development in use, of IT, and the need for models and methods for sustainable, distributed co-constructive design processes.

Keywords

Design in use, evolutionary design, shop floor IT management, public services, one-stop shops

INTRODUCTION

'*What, precisely, do you mean by design?*' the moderator asked us. Five of us, two researchers and three Ph. D. students, had just finished constructing a collage on the whiteboard. We were cooperating around the *Design of IT in Use* project, which focuses on supportive technologies for public service provision. Each co-presenter had added more steps and more detail to the multi-colored complexity of circles, arrows, figures and keywords we were using to illustrate our presentation.

The design we were talking about had gradually shifted character as we worked our way across the whiteboard, from left to right, from systems developers and consultants to local technicians, to web designers, service providers and 'citizens/users'. On the left-hand side of the board, design seemed to be about the solution-planning phases of software engineering. Design, here, was the work necessary to get done before, and during, the actual coding process. On the right-hand side of the board, a social constructionist perspective prevailed. Here, *use* of technology was *design* of technology. The moderator's question brought our varying perspectives and starting-points into sharp focus. *Who is designing what for whom?*

Different groups of people were definitely cooperating in design processes in this panoramic view of IT in public service provision. In fact, the software developers seemed to be playing a rather minor role in the over-all picture. A participatory designer, or a software developer practicing participatory design – the normal addressee of participatory design methods – would miss a large part of the cooperation around appropriation and tailoring in this picture, as it was taking place beyond the temporal and organizational scope of any project.[1]

In *PDC 02 Proceedings of the Participatory Design Conference,* T.Binder, J.Gregory, I.Wagner (Eds.) Malmö, Sweden, 23-25 June 2002. CPSR, P.O. Box 717, Palo Alto, CA 94302 cpsr@cpsr.org
ISBN 0-9667818-2-1.

[1] See [20] as an example concerning the presumption of the participatory designer as the addressee of participatory design methods, and the project as the scope in which participatory design takes place.

In this article we explore this 'participatory design in the wild' further, and discuss its implications for the conceptual framework of participatory design as well as for its methods and tools. Suchman challenged the borders between technology production and use as culturally established and often ideological rather than empirical. [30] With our observations, we would like to contribute to a better understanding of evolving 'working relations of technology production and use', and of the implications for the conceptual frame of participatory design as well as its methods and tools. We indicate in which directions participatory design might be developed to accommodate and support practices of design in use such as we have observed in our field studies.

Design as a continuing process that goes on after the formal end of the software development project is, of course, 'old news' [24, 26]. Tailoring and the role of end-users in customizable software has been taken up in a diversity of scientific discourses. The 'new news' is, that this is where much of the action is today, and it is a much more complex and diverse scene than it was ten years ago. In this article, we argue that the framework and rhetoric of participatory design, at least from a Scandinavian perspective, need to be expanded to include and highlight on-going design in use and evolving practices of 'PD in the wild'.

DESIGN IN USE

In a joint, interdisciplinary research project, we started to use the term 'design in use' [7, 18] to capture practices of interpretation, appropriation, assembly, tailoring and further development of computer support in what is normally regarded as deployment or use. In the following, we present examples of these practices as a base for discussing the implications of such changes in the use of technology for the understanding of design and use, and for the understanding of participatory design in such contexts. Not only the initial embedding of an application into a specific context, but also the practices that develop around tailoring, can be considered as 'design in use'. Understood in this way, 'design in use' becomes a powerful concept, which highlights the incompleteness of any technical artifact, even the most skillfully designed one, and the need for its continual adaptation and further development. 'Design in use' emphasizes the creativity that lies in the embedding and use over time of the technical artifact. It therefore both motivates the conscious implementation of participatory design, and highlights what often passes un-noticed in design discussions: *the actual on-going cooperation in design practices in everyday use of technical artifacts.* [2]

In the following section, we present different practices we have observed in our case studies. These examples provide the base for our claim that, in many cases, design and use should not be regarded as two separate and sequential activities, but rather as on-going in parallel, intertwined, overlapping, with shifting foci and agencies. The question this raises is: how might these different, co-existing practices of design be more deliberately and consciously put in dynamic relation to each other?

The relationship between design and use, their overlap and intertwine, is a core issue of participatory design. Use has to come to design, and design has to come to the users, in order for developers and users to cooperate around the design of software or other technology.

Since the involvement of users in design, development and introduction of technology was a novelty, the developing participatory design community and related research discourses addressed a variety of issues: from organizational politics, via methods to mediate the cooperation across professional borders, to an adequate organization of the development process to provide space and time for such cooperation. [5,21] The majority of these methods aim at bringing use to design.

Parallel with the development of methods to bring use to design, the understanding developed of what happens after the introduction of a new computer application. Reports from detailed studies of computer supported cooperative work challenged the naive understanding of use as the implementation of its anticipation in design. Rather, these studies showed how software is embedded into a developing use context in a creative way. It is interpreted [8], used in unanticipated ways [28], adapted [32] and appropriated in relation to diverse work practices. If the 'clay of computing' is understood as the computer application as it is used [7], the creative and conscious embedment can be called design as well. 'Use is design' was the radicalization of this understanding [1].

As a more complex understanding of use developed, the support for flexibility in use became an issue for Human Computer Interaction. The possibility of object oriented, graphical user interfaces, the desktop metaphor, the tools and materials metaphor, [12] designing props to be used when necessary [4] – all these design concepts aim at supporting the user while leaving freedom for localization and appropriation, for situated use. Tailorability has become a subject across scientific communities. [22, 18, 25, 11] These techniques allow for bringing design, in a more conservative understanding, into the use context.

[2] The concept of 'design in use' as we discuss it here has links to the distinctions made by Argyris and Schön in *Organizational Learning* [2] between espoused theory and theory-in-use. Due to the lack of space, we cannot explore this interesting connection further in this article.

With the development of computer applications to support cooperative work, and, even more, with the use of networked applications as an infrastructure for data centered business and administration, the relationship between design and use seems to shift again. Work and business practices – in our cases, the provision of services – are intrinsically related to the supporting technical infrastructure. The evolution of the Internet (and various local and regional intranets) has changed the scene even more, blurring the boundaries between content and form, and, therefore, between users and designers.

Participatory design has to be taken beyond the scope of a normal development project. After presenting and discussing practices of design in use we observed in a joint research project, we conclude with what we see as important implications for participation of different actors in the on-going co-development of service provision and supporting technology, and the methods and techniques to mediate this co-development.

EVOLVING PRACTICES OF DESIGN IN USE

We have studied use and design of supportive technologies for public services. These studies have been carried out as part of a three-year research and development project, *Design of IT in use - supportive technologies for public services* (DitA)[3]. Development of provision of public services is intrinsically related to the further development of supporting and enabling infrastructure. Service provision and computer support are so intertwined and interdependent that the development of one implies the further development of the other.

The interdisciplinary cooperation between the department of Human Work Science and Media Technology and the department of Software Engineering and Computer Science, within the framework of this project, allows us to relate organizational, work practice, and technical development and participatory design without unduly reducing the complexity and tensions between these various perspectives. Using ethnomethodologically informed ethnographic field studies, including methods such as observation combined with video-recording as a tool for detailed interaction analysis, we are studying work practices of service providers and the cooperation between service providers and software developers designing and supplying applications for service provision. This methodology we combine with an action research oriented Scandinavian approach, using various methods such as workshops together with users, in order to support and ensure user participation in the design of public services on-line.

In this section, we take a closer look at practices of design in use in two of the municipalities in the DitA project, Ronneby and Sölvesborg[4]. The practices we present show concrete examples of the co-development of software, organization and work practices. In their concreteness, they raise a number of issues, which we will return to in the section thereafter.

In Ronneby; designing sets of possibilities

Ronneby was one of the first Swedish municipalities to launch its web site on Internet (in the spring of 1995). Since January 1999, the municipality has run its own intranet, with links to information on the public web. Routines are being introduced to allow all departments to be responsible for publishing their own content and services. During our study, the municipality had chosen to successively implement *Intrainfo*, a platform for among other things an administrative application for Internet/intranet publishing. This application was being co-developed by representatives from the municipal information department and a systems developer from the software consultancy firm, TietoEnator, that was developing and marketing *Intrainfo*.

Most of the fieldwork in Ronneby was done by two third-year students of the MDA program,[5] who were writing their bachelor's thesis about local IT design in the municipality. [13] The study focused on a part of the application called *Intradok*, a document-management system which contains templates, that is, sets of rules for writing, displaying and storing various types of digital documents within the municipal organization and publicly. During a period of several months in the spring of 2001, we studied the on-going design and tailoring of *Intradok* to support the publishing of the municipality's personnel advertisements on the Internet and the municipal intranet. Here, we identified different levels of cooperative design practices.

Consultant and local co-designer cooperation

First, from our point of view, there was the design work going on in close cooperation between the consultant/sys-

[3] Funded by the Swedish Agency for Innovation Systems VINNOVA, April 2000 - December 2002 (project no. 2001-03659). The partners are five municipalities, two software consultancy firms, a Call Center and us researchers at the Blekinge Institute of Technology.

[4] In reporting from this project, we have jointly agreed to use the real names of the involved partner organizations, but fictive names for individual persons in the different cases.

[5] The MDA program is a master's program at Blekinge Institute of Technology where Work Science and Computer Science are combined in educating future systems developers. MDA is the Swedish acronym for *People, Computers and Work*.

tems developer, Martin, and the chief of information in the municipality, Lena. Martin mainly worked at his home office in a city some 350 miles northwest of Ronneby by car. Thus, although he visited the municipality once a month on an average, most of the cooperation with Lena was carried out by phone, supplemented, if necessary, by the use of *pcAnywhere™*, an application for accessing and temporarily taking over control of a local network-based PC application from afar.

The telephone and computer communication sessions turned out to be teaching and feed back sessions at the same time. For example, Martin showed Lena how to set up a new template for a specific set of documents, together with the related rules for access and editorial rights. Going through the possible choices, Lena wanted to be able to assign the same rights to people with different organizational roles, that is, connect several roles to one access profile. She gave several concrete examples of when and why this might be necessary. As this was not possible to do in the existing application, Martin made a note to allow for it in the next version. He had not taken into account the organizational structures and practices motivating this need until Lena pointed them out to him. Expertise shifted back and forth between them during this mutual learning process.

Martin's highly evolutionary way of developing the software enabled him to flexibly react on requirements from the few pilot users spread out over several municipalities. Lena was one of them. She appreciated the co-operation and was enthusiastic about being able to contribute to a useful tool for her organization. Some other municipalities who did not co-operate as closely with Martin had difficulties with the frequent changes and delays caused by his unconventional development practice.

Local analysis and reorganization inform on-going design

The organizational analysis and design work was going on locally in different departments. The work which was to be supported by the new system was being discussed and, to some extent, reorganized. This was seen not only as reorganization of work processes to fit new technologies, but as analysis work necessary for informing the on-going design and development of *Intradok* in order to fit it to the work organization in Ronneby. Here, besides the chief of information, Lena, there were three personnel secretaries in the municipal organization involved, Anna, Eva and Marie.

Until now, the normal procedure had been that Anna, Eva and Marie received the information about help wanted within, and wrote the advertisements for, their three different functional areas in the municipal organization. They then sent these ads via e-mail to Lena, who edited them and published them on the Internet/intranet. With the implementation of *Intradok* in the organization, this would be changed, such that the personnel secretaries could in future do their own editing and publishing of digital documents.

During a participatory workshop, the three personnel secretaries discussed their current routines, compared their different ways of working, and designed new routines with the help of *Intradok*. As they had never before compared their practices, they regarded these discussions about different ways of working as useful and rewarding. The participation in the tailoring of the new tool allowed them to reflect and consciously change their own routines and practices.

Who is designing what for whom?

Who, then, was the designer of the Ronneby version of *Intradok* being developed? The consultant, Martin, was designing an application to fit the needs of Ronneby municipality. But this application was being deliberately developed to afford and support future local design and development of the technological infrastructure in the municipality. The municipal chief of information, Lena, in cooperation with Martin, was designing an organizational infrastructure and delimiting and creating future design space for public services on the Internet and the local municipal intranet. The three personnel secretaries, Anna, Eva and Marie, were designing templates and a standardized framework for digital municipal employment ads and policies, making use of the space that Lena rigged and delimited for this purpose in *Intradok*. Looking in the other direction, we see that feedback from the local design processes in Ronneby was informing the overall design of the generic application *Intrainfo*, of which *Intradok* is a part.

Is this participatory design?

In what sense were we seeing participatory design in Ronneby? Mainly in the cooperation between the municipal chief of information, Lena, and the consultant, Martin. They carried out a number of design sessions together, usually by phone and with the support, when necessary, of the remote access and control facilities of *pcAnywhere™* . When it came to the design work of the three personnel secretaries, ideas and experiences were shared and explored amongst them during the workshop. This was the first time they had discussed their ways of organizing their work together, even though their offices were next door to each other. They were surprised to discover how differently they worked, and realized they could learn a lot by sharing experiences and ideas.

Today, Ronneby municipality is developing an organizational structure to support decentralized participation in design, adaptation and use of the common information infrastructure. The successor of Lena (who has moved to another job) is building up a committee of what can be called 'shop floor IT managers' in different departments, so that local interests can be considered and so that as much development and implementation of the IT infrastructure as possible can be delegated to the individual departments that will use it in their everyday work practice.

In Sölvesborg; "The designers? That's us!"

In Sölvesborg, we have been studying the use and development of front-office computer support in a public service one-stop shop (*medborgarkontor*, in direct translation 'citizens' office'). The one-stop shop is centrally located, in the entrance of the town hall. It offers information and services to citizens of the municipality as well as to visitors from out of town. A team of public service guides staff the one-stop shop, answering questions and helping out, either face-to-face, by phone, via e-mail or, in some cases, by internal or regular mail. The one-stop shop team is responsible for keeping much of the municipal information on the Internet and the municipal intranet updated, and for the further development of public information and services on the municipal website. The main motivation for this is that they are well acquainted with what kinds of information people ask for and need. As they use the Internet/intranet themselves all the time on the job, they are aware of design and accessibility issues.

During our field-studies of IT in use in the one-stop shop, we researchers implied to the head of the one-stop shop, Helena, that we would like go a step beyond the front-office staff, and talk to some of the technicians and systems designers. '*The designers? That's us!*' was her immediate and spontaneous reply. For a group of researchers aiming to use a participatory design approach throughout our research and development project, this was an extremely inspiring answer to get from the would-be 'participant' side. But it was also a serious challenge to our current worldview. On the one hand, it indicated the users' active and responsible involvement in on-going IT development processes in the municipality. On the other hand, it became obvious that we were dealing, here, with several different understandings of what it means to design technical support for front office work.

Choosing what applications to buy

On the level of choice of what applications to buy, install and use, the work team in the one-stop shop in Sölvesborg has developed a pragmatic approach. In 1992, when the one-stop shop was inaugurated as one of the first of its kind in Sweden, most of the applications accessible via the municipal network were mainframe systems, supplied by the main national dealer in software for municipal administration at that time. Today, the front office team uses the Internet/intranet, regularly accessing and using more than 20 different applications from 17 different software providers[6]. One of the most recent acquisitions is the new computer support for the telephone exchange. This application was chosen from a number of offers because the supplier was the only one who not only promised that the application could be usefully integrated with the existing computer support, but was actually able to prove it in practice. In the purchasing process, the front office work team set up the main functional requirements and, in the end, made the final choice between different available systems. The municipal technicians gave them advice about technical aspects of the various systems offered, based on the overall IT strategy in the municipality, network capacity etc. The front office team has gained considerable experience during the past few years in specifying their needs and purchasing necessary applications to further develop their computer support. At the level of analyzing needs, exploring options and deciding on what new pieces to add to the puzzle of different applications in use, their claim of being 'the designers' seems indeed to have substance.

Designing the services along with the technical support

Beyond that, shifting our main focus, for just a moment, from software to service provision, it seems that the front office work team is deeply and daily involved in designing and developing public services. Here, too, they have a point in saying '*The designers? That's us!*'

Taking part in design through user feedback processes

But even narrowing our focus to software design of specific applications, we have found that there are interesting things going on in and around the one-stop shop in Sölvesborg. One of the most frequently used applications is a niche application for booking locales (tennis courts, conference rooms etc.), an application which has been developed by a small software consultancy firm in the region. When asked what parts of the current computer support they find most useful, functional and well designed, this is the application the front office team uses as the best example. The consultancy firm keeps in contact with their customers, municipalities and associations all over Scandinavia, and provides support both via telephone, via their website on the Internet, and via visits. They have customer support meetings, where problems and new ideas are discussed and prioritized, between 8 and 10 times a year. The processes of continual support, take-up on customer feedback and further design and development, which this firm cultivates, may well be a large part of the reason for their successful product. They produce approximately 15 – 20 new versions of their basic application per year. These are continually being provided to all customers via the firm's website, with descriptions of 'what's new'. This allows their customers to choose for themselves whether the newest version is one

[6] At the time of our latest count, and not including all the more or less invisible middleware that keeps the network going. Basically, we counted the program icons on their digital desktop, and checked that these were what they themselves perceive as the applications they use.

they need to download or not, depending on what new functionalities have been added.

Through the customer feedback processes, there is apparently some substance to the Sölveborg front office team's claim of being their own designers, even here. Admittedly, they represent only one of some 250 customers giving continual feedback about the product. However, it is clear that they themselves feel that they have been able to act as co-designers in the case of their most appreciated application, and that they still have a co-constructive role in its continued development.

Recent development in Sölvesborg has lead to the font office team earning a more official status in the organization as local designers of the municipality's intranet. They are consultants for other departments in how to use the existing possibilities, discussing and coordinating improvements.

ISSUES RAISED BY DESIGN IN USE

The practices we have described here reveal a shifting responsibility for the observed design in use. As agency shifts, the object of design changes character as well: the supporting software, its adaptation, the whole infrastructure for service provision. The computer applications become a boundary object [29] for multiple design practices/activities. Some of the observed practices look much like tinkering [9], that is, in an ad hoc way making heterogeneous applications work together. However, there is practice- and experience-based deliberation and judgment behind both what is in use and what is considered necessary to implement next, as we have seen in the Sölvesborg case. The sense of ad hoc-ness lies, rather, in the fact that observed practices seem often just to take place, unaccounted for, not supported by the formal organizational structure or allotment of resources.

Would an organizationally defined function and structure for 'shop floor IT management' [14] be a way of making these design and development activities, and the multiple and shifting foci of design they represent, more visible and organizationally legitimate? Something to this effect seems to evolving, in different ways, both in Ronneby and in Sölvesborg. If so, what should professional 'shop floor IT management' look like? What does such an overlap and intertwine of use, design in use, maintenance and development require of software development? We can design software in ways so it can be easily adapted to certain changes in work practice. How does that influence the distribution of design work between computer professionals and different groups of local co-designers/web designers/ service providers/users/citizens?

As we explore these topics in the following, we deliberately do so as researchers coming from two different research areas. Two of us have a background in Software Engineering and Computer Science, and thus focus mainly on software development processes, models and methodologies for use-oriented design of software. The third has a background in Work Science and Informatics, as well as work-life experience from white-collar office work and IT consultancy. Jointly, we are striving for a constructive dialogue between different standpoints concerning design of IT in use. At the end of the section, we bring together the challenges we see for participatory design in the evolving practices we have been observing.

Shifting Foci of Design

What, precisely, we mean by design, then, depends on from what standpoint we are viewing information technologies. Traditional systems development and computer science perspectives take a narrow view of design, defining it as the modeling process between requirement analysis and the actual coding of the computer application. Newer programming methodologies have stretched the concept of design to include the coding process, and evolutionary design is often understood as including the entire lifecycle of the application [15]. Even in evolutionary design, however, the *active agents* of design are normally seen as being professional software developers or participatory designers, situated within a systems development community of practice – usually a software consultancy firm, or an in-house IT department.

What we have observed during our field studies in the DitA project indicates that there are other people out there, in a number of different communities of practice, who are actively involved in the design of various aspects of the computer support for public service provision. From the point of view of the municipal service providers, whose work practices we have been observing, there are a large variety of designers involved in the development of their computer support. There are local technicians designing and managing the technical network. There are local co-designers cooperating with consultants and/or in-house municipal IT departments in developing and customizing various web-applications. There are, among the front office service providers themselves, as well as elsewhere in the municipal organization, local web-designers designing and maintaining web-sites, design work that often includes modeling and developing shared on-line databases, templates etc. And, as we saw in the one-stop shop in Sölvesborg, front office service providers are also, in some cases, designing their own computer support on a meta-level, having been given the authority and responsibility of choosing what applications to buy and make use of.

As the focus shifts, we see, beyond 'pure' software design, a number of on-going design practices involving IT in use. We see services being designed, on-line or off, but usually involving IT, in one way or another. We see constellations of different off-the-shelf applications being designed, individual applications being further developed and tailored, and whole technical infrastructures thus being designed and developed in a distributed, yet often locally cooperative way.

Who is responsible for these design processes, and for taking care that they are carried out in a cooperative way?

Design for Change

The possibilities for users' participation in design, and for the continuation of design, once a device is taken into use, depends not only on organizational circumstances, but also on the technology developed and used. Because of its malleability, for example, software allows for evolutionary prototyping for the integrated co-development of design and use. Still, to accommodate different people in different roles and with different foci appropriating the software to the developing work practice and service provision requires a change in perspective also in software development and design. Software development is often perceived as producing a 'solution' for a 'problem'. A 'design for change' perspective puts the support of work practices in focus, work practices that are expected to change in anticipated as well as in unexpected ways.

Software that is part of a complex technical infrastructure, like in the cases presented above, has to be designed in such a way as to allow for design in use. The goal is no longer a fitting application for a specific use context, but software that can be related to different work practices and service provisions, and that can be adapted if and when necessary.

Taking change into account from the beginning requires a different way of thinking. Changes and different kinds of use practice have to be anticipated. Different techniques to provide for adaptability have to be evaluated with respect to the specific circumstances at hand. What adaptations can and should be done by which users? What role can a local developer or a systems administrator take over? How to provide for easy maintenance? Here, use, development and technical contexts, as well as the interaction between them, have to be taken into consideration [11].

Today's design methods offer very little guidance concerning how to provide space and support for the cooperation of users, local developers, and computer professionals in on-going design. Even participatory design methods primarily focus on fitting support for specific work practices, rather than design for change.

Shop Floor IT Management

During earlier field studies of work practices in one-stop shops, carried out in 1992 – 1996, before the Internet had had any real impact on the computer support in use, we found that there was a sizable gap between declared IT strategies – which were often very visionary – and existing organizational infrastructures for supporting everyday use of IT in the municipal organizations we were studying [14].[7] In order to bring this gap into focus in on-going strategic discussions about IT on a managerial level, as well as in our research work, we began to talk about *the art* of *shop floor IT management*. Shop floor IT management, in this context, was seen as the everyday work of making IT work, that is, the mundane, on-going problem-solving, tuning, tailoring, further development and design in use of the existing computer support, and the integration of new applications into this existing environment. Ideally, IT in use in an organization with a well developed IT strategy would be a self-lubricating and self-enhancing system, with the feedback from users directing the work of shop floor IT management.[8]

At that time, we located the responsibility for the gap we detected between stated IT visions and actual IT use (and misuse, or non-use) with management and in-house IT departments in the municipalities we were studying. Here, we felt, there should be technicians and local designers involved in help-desk and tailoring activities, supporting the adaptation of IT in the everyday work practices throughout the organization. This was an infrastructure that seemed to be largely lacking.

In our more recent field studies, carried out during 2000 and onward (the DitA project is still on-going), the issue of shop floor IT management is still highly relevant. There is still a gap between officially stated managerial IT strategies and what is actually being accomplished with IT within the organization. However, when it comes to IT design agency, the situation in the municipal organizations we are studying is more diverse and complex than it was before the Internet, and various intranet solutions, became common. As we have shown with the examples in this paper, today there are design activities going on in many different parts of the organization, involving many different categories of users as active designers/co-designers. Technicians, web-designers, service providers – there are many people, and groups of people, involved. Thus, in some sense, shop floor IT management is actually going on, right out there on the shop floor. But it is not officially accounted for, nor is it supported by organizational and technological infrastructures, and given resources, in proportion to its apparent

[7] These studies were mainly carried out as part of the research project *Working at the front – skill, cooperation and computer support in public service one-stop shops*, which was financed by the Swedish Council of Work Life Research (project number 94-0349).

[8] Similar participatory and cultivating approaches to IT in use have been explored for by others. [22, 8,17, 32]

relevance for smooth everyday use of IT in getting the day's work done.

Reconsidering Software Processes

The described interlacing of design, development, use and purchase requires new and alternative ways of organizing software development. In both case studies, the practice of the software developers involved has been unusual if not contradictory to what is considered state of the art in software engineering textbooks.

The highly iterative and cooperative development practices with close feed back cycles between Lena and Martin in the Ronneby case ensured that the spaces for adaptation are designed to accommodate municipal practices. According to software development standards they are anarchistic and unmanageable. The development practice at the favorite software provider of the one-stop shop staff in Sölvesborg is another example. The close interaction with the users that results in frequent versions that can be downloaded if needed, is not accounted for in the organization of customer developer relationships as taught at university courses. But would a software project starting with a defined set of requirements have been able to accommodate the co-development of technical infrastructure and service provision we see here?

Evolutionary software process models like STEPS [15] can capture part of the initial dynamics of the introduction of new software. They have been criticized, as being not applicable in commercial settings as there is no previously defined project goal. Extreme Programming, developed as an answer from practitioners to more and more rigorous project management strategies, now proposes similar ideas: tight iterative releases that can be taken in use from the very beginning, the possibility to adjust the requirements and project goals after each release, and an end user on site. [3]

The observed practices of design in use seem to ask for even more radical re-conceptualizations of software processes. New developments of applications intertwine with use, maintenance, tailoring, adaptation and further development. To focus on an isolated controllable development process and regard the rest as - not so important - operation and maintenance, ignores the changing way in which software is implemented and used. (See also [6])

Ways to coordinate and manage such patchworks of design activities as we were able to observe in our case studies are still to be developed. Understanding software design as networks of decisions in relation to use, technical and development contexts [16] can provide a starting point. A more realistic way might be to understand development tasks, design in use and use as parallel activities with shifting intensity and shifting main actors. This would allow for maintaining continuity in co-operation between minor tasks that can be handled as part of everyday operative support and bigger chunks of development of maintenance work organized in projects and under the leadership of software developers. The coordination of these networks of activities can only be achieved in close continuous cooperation between users and developers.

PARTICIPATORY DESIGN BEYOND THE PROJECT

The when and where of participatory design has often, explicitly or implicitly, been set by the software design process, which is usually run in project form.[9] Agency, roles and terms of responsibility have tended to be defined within this framework. Although software design projects are undeniably an important arena for participatory design, the examples given above indicate that cooperative design is actually going on in a lot of other contexts as well.

By starting with a focus on existing work practices and IT in use in a number of specific work places, rather than on a software development project, we gradually came to change our understanding of design, and, consequently, to question some of the assumptions underlying the basic framework and methodologies of participatory design. If design is seen as continually on-going in many different locations and forms, and intricately interwoven with use, rather than as primarily contained within software development projects, this raises several important issues for participatory design.

Within the software development community, it highlights software flexibility, adaptability and sustainability – design for change. It points towards the need for reconsidering software design processes, and developing methods for linking them more closely and continually to user feedback and rapid and efficient version management, for example. But both within and beyond the software development community, it brings into focus issues of coordination between use, design in use, adaptation and development.

From a broader organizational perspective, there are practically no formally visible infrastructures or resources for what we call shop floor IT management, that is, support for local adapting, tailoring, tuning, and continual design and development in use of IT. Focusing on shop floor IT management raises issues about how to make the on-going 'participatory design in the wild' more visible, and give it support within the existing organization, for instance in the form of standards, methodologies, modeling languages and other means of representation for cooperative development

[9] The primarily project-oriented focus of participatory design may well be more pronounced in Scandinavia than elsewhere. It has been questioned previously by researchers in Britain and North America, in responses to the 'Scandinavian Challenge'. [23].

and – not the least – personnel resources. Surely, methods from participatory design could be more consciously and deliberately explored, adapted for, and applied in, these contexts.

Participatory design has hitherto mainly been an issue for the software development community and, traditionally, in Scandinavia at least, the workers' unions. Today, more than ever, we believe it is an issue for everyone. Practices of design in use challenge traditional concepts of designer/user roles as well as concepts of what it is that is the object of design, and in what contexts design takes place. To whom should we teach our methods, and what can we learn in the process?

CONCLUSION

We have gone out into the field to study IT in use. In so doing, we have gradually become aware of what we have called, provocatively, 'PD in the wild'. Modern IT networks, with a variety of applications from different software providers, new web-design tools, and the integration of customization processes with on-going version management, are just a few of the developments which appear to be providing a base for the evolving practices of design in use we have observed.

When Hutchins challenged cognitive theory by using anthropological methods to explore 'Cognition in the Wild' [19], he was challenging a whole conceptual framework. We started, more tentatively, by asking ourselves what our findings might mean for methods and practices of participatory design. But we found, as we explored these issues, that we were in some sense beginning, like Hutchins, to challenge the framework. Others have challenged participatory design to move beyond a certain set of organizational structures, claiming that traditional participatory design doesn't accommodate small companies, or networked organizations, for instance [27, 31]. In this article, we have attempted to broaden the when, where, and how of design, and take participatory design beyond the software development project. We have explored how we can bring a multitude of different perspectives on design, and a multitude of different kinds of design and designers, into the overall picture and understanding of on-going design and development of IT and – in our case – public services.

Participatory design is, as we see it, no longer primarily a professional issue for software developers, but has to be extended to the relationships between different user-designers, and, beyond that, between them and their clients/customers/service-seeking citizens in general. As technological and organizational infrastructures change, participatory design has to change, too. Within the DitA project, we are now exploring these issues further in two directions. On the one hand, we are studying the local cooperation between web-designers and service providers, and between citizens and service providers, to see how participatory design methods can be adapted and used in these contexts. On the other hand, we are taking a closer look at 'unorthodox' methods of participatory design in use between systems developers and users. Thus, the field studies at the consultancy firm that has developed the favorite application of the front office team in Sölvesborg are still on-going, and here we are now looking more specifically at issues around *designing for* design in use, i.e. processes and methodologies for supporting use-oriented design of flexible and adaptable systems.

Issues we are continuing to explore jointly and in depth in our interdisciplinary research cooperation around IT and public services are; means of representation for participatory co-development of services and technology, issues of accountability, methods for on-going participatory design, design for change, and new ways to integrate design, development and use of technology. In one way or another, these questions all have to do with developing sustainable organizational support for 'PD in the wild', that is, domesticating and cultivating participatory design in everyday use of IT.

As for the initial question in our introduction; '*What, precisely, do you mean by design?*' we don't really expect we will ever be able to agree on one answer.

ACKNOWLEDGMENTS

Students from the *People, Computers and Work* program at the Blekinge Institute of Technology have carried out projects in connection with our research. Thanks especially to the practitioners in Ronneby, Sölvesborg and TietoEnator, whose work and cooperation we were allowed to study at close hand. Tone Bratteteig helped us problematize the concept of design. And thanks to the reviewers whose constructive critique helped us in developing our arguments.

REFERENCES

1. Allen, C. Reciprocal Evolution as a Strategy for Integrating Basic Research, Design and Studies of Work Practice. In Schuler, D. and Namioka, A. (eds.), *Participatory Design: Principles and Practices*. Lawrence Erlbaum Associates, Hillsdale NJ US, 1993, 239-252.

2. Argyris, C., Schön, D. A. *Organizational learning: a theory of action perspective.* Reading, Mass. 1978.

3. Beck, K. *eXtreme Programming explained. Embrace Change*. Addison-Wesley 2000.

4. Binder, T. Intent, Form, and Materiality in the Design of Interaction Technology. In Dittrich, Y., Floyd, C. and Klischewski, R. (eds.), *Social Thinking – Software Practice*. MIT Press, 2002, 451-468.

5. Bjerknes, G. and Bratteteig, T. User Participation and Democracy: A Discussion of Scandinavian Research on

System Development. *Scandinavian Journal of Information Systems* 1995 (8), 73-98.

6. Braa, K., Bratteteig, T., Øgrim, L. Organising the redesign process in system development. *Journal of Systems and Software, Special issue on Information System Development*, May/June 1996

7. Bødker, S. *Computer applications as mediators of design and use – a developmental perspective*. DAIMI PB-542, Computer Science Department, Århus University, October 1999.

8. Christiansen, E. in collaboration with Draper, P. and Mathison, A., *A Gardening Attitude. A Case Study of Technical Support Work*. Institute for Research on Learning, Menlo Park, CA, 1996.

9. Dahlbom, B. and Mathiassen, L., *Computers in Context. The Philosophy and Practice of Systems Design*. NCC, Blackwell, Oxford UK, 1993.

10. Dittrich, Y *How to make Sense of Software – Interpretability as an Issue for Design*. Research Report 19/98, University of Karlskrona Ronneby 1998.

11. Dittrich, Y. and Lindeberg, O. Designing for Changing Work and Business Practices. In Patel, N. (ed.) *Evolutionary and Adaptive Information Systems*. IDEA group publishing (forthcoming).

12. Ehn, P. *Work-oriented Design of Computer Artefacts*. Arbetslivscentrum, Stockholm, Sweden, 1988.

13. Ekstrand, S., and Hansson, C. *Design och utveckling av IT-verktyg – ger ringar på vattnet i en organisation* Master Thesis, Department of Human Work Science and Media Technology, Blekinge Institute of Technology 2001.

14. Eriksén, S. *Knowing and the Art of IT Management. An inquiry into work practices in one-stop shops*. Ph.D. thesis, Department of Informatics, Lund University, Lund, Sweden, 1998.

15. Floyd, C., Reisin, F.M. and Schmidt, G. STEPS to Software Development with Users. In Ghezzi, G. and McDermid, J.A. (eds.), *Software Development and Reality Construction*. Springer Verlag, Berlin, 1989, 48-64.

16. Floyd, C. Software Development as Reality Construction. In Floyd, C., Züllighoven, H., Budde, R., and Keil-Slawik, R. *Software Development and Reality Construction*. Springer Verlag: Berlin 1992.

17. Gantt, M., and Nardi, B. A. Gardeners and gurus: Patterns of co-operation among CAD users. In: *Proceedings of the CHI '92* , ACM, New Yorck 1992, 107-117

18. Henderson, A. and Kyng, M. There is no place like Home: Continuing Design in Use. In Greenbaum, J. and Kyng, M. (eds.), *Design at Work*. Lawrence Erlbaum Associates 1991, 219-240.

19. Hutchins, E. *Cognition in the Wild*. MIT Press, Cambridge, MA US 1995.

20. Kensing F. Participatory Design in a Commercial Context - A Conceptual Framework. In *Proceedings of the PDC, Nov. 28- Dec. 1, 2000, New York, USA*, 116-126.

21. Kensing, F. and Blomberg, J. Participatory Design: Issues and Concerns. *Computer Supported Cooperative Work* 1998 (7), 167-185.

22. Maclean, A. Carter, K., Lovstrand, L. and Moran, T., User-Tailorable Systems: Pressing the Issues with Buttons. In *Empowering People: CHI'90 Conference Proceedings*. Seattle WA:ACM, 1990, 175-182.

23. Muller, M., Blomberg, J. L., Carter, K. A., Dykstra, E.A. Halskov Madsen, K., Greenbaum, J. Participatory Design in Britain and North America: Responses to the Scandinavian Challenge. *Proceedings of the CHI '91. Reaching through Technology*. New Yorck: ACM Press, 389-392

24. Mumford, E. The Participation of users in Systems Design: An Account of the Origin, Evolution and Use of the ETHICS Method. In: Schuler, D. and Namioka, A. (eds.) *Participatory Design. Principles and Practices*. Hillsdale New Jersey, Erlbaum Associates 1993, 257-270.

25. Mørch, A. and Mehandjiev, N. Tailoring as Collaboration: The mediating Role of Multiple Representations and Application Units. *Journal for Computer Supported Cooperative Work* 2000 (9), 75-100.

26. Orlikowski, W. Learning from Notes: Organizational Issues in Groupware implementation. Proceedings of the CSCW 92, New Yorck, ACM Press 1992, 362-369.

27. Robertson, T. Participatory Design and Participative Practices in Small Companies. Blomberg, J., Kensing, F., and Dykstra, E. A. Proceedings of the PDC '96. CPSR, Palo Alto CA 1996, 35-43.

28. Robinson, M. Design for unanticipated use... In De Michelis, G., Simone, C. and Schmidt, K. (eds.), *Proceedings of the Third European Conference on Computer-Supported Cooperative Work,* 13-17 September, 1993, Milan, Italy, 1-16.

29. Star, S. L. The Structure of Ill-Structured Solutions: Heterogeneous Problem-Solving, Boundary Objects and Distributed Artificial Intelligence. In Huhns, M. and Gasser, L. (eds.) *Distributed Artificial Intelligence II*. Morgan Kauffmann, Menlo Park 1989, 37-54

30. Suchman, L. Working Relations of Technology Production and Use. *Journal of Computer Supported Cooperative Work* 1994 (2), 21-39.

31. Törpel, B., Wulf, V. and Kahler, H. Participatory Organizational and Technical Innovations in Fragmented Work Environments. In Dittrich, Y., Floyd, C. and Klischewski, R. *Social Thinking – Software Practice.* MIT Press 2002, 331-356.

32. Trigg, R. and Bødker, S. From Implementation to Design: Tailoring and the Emergence of Systematization. In *Proceedings of the CSCW '94.* ACM-Press, New York 1994, 45-55.