Scenarios as springboards in design of CSCW

Susanne Bødker & Ellen Christiansen***
*Department of Computer Science
Aarhus University
Ny Munkegade
DK-8000 Aarhus C

**Department of Communication
Aalborg University
Langagervej 6, box 159
DK-9100 Aalborg

Abstract

Our point of departure is that the relationship between social science, cooperative working and technology is not as much a matter of differences in understanding, as it is a matter of how to accomplish change. This chapter outlines an approach to design of CSCW where change is addressed in terms of *expansion of the work practice*. To facilitate the change process as a process of expansion, scenarios are used as springboards. Creation and use of scenarios are supported by a conceptual *toolbox*. The foundation for this toolbox is an understanding of the design process as *abductive thinking* consisting of idea generation and systematic reflection, and an understanding of design tools inspired from activity theory. As design processes may involve different communities of practice, we discuss the role of scenarios as boundary objects.

Introduction

In this paper we shall primarily address the discussions of social science, cooperative working and technology from the point of view of practical design. The constitution of the design situation and the interplay between design and use is an important yet often neglected topic, and despite the fact that design of computer artifacts inevitably implies change, designers are given little help to consider and reconsider the outcome of creating something new, as far as the social setting is concerned. They live on the borders between several communities of practice, surrounded by conflicting interests and requirements, and find themselves caught in a dilemma between awareness of tradition and orientation towards transcendence: on the one hand starting out from the praxis and history of the users in question, on the other hand making sure that something qualitatively new gets shaped in the process. A dilemma which is also reflected in the seemingly contradiction between abstract theoretical and situated practical understanding, between using a framework or a description method to structure the analysis of the situation, and an open-minded "letting the situation speak to you".

Our background for this undertaking is multi-disciplined. Our main concern for many years has been for understanding systems design processes in general, and user participation in particular. We have taken part in various action-oriented research projects to deal with this issue, and we have tried to work theoretically from within human and social sciences as well as technically. Part of this background includes a general rejection of systems descriptions as *the* central design tool, replacing them with more experience-oriented devices such as prototypes. Despite the fact that design of computer artifacts inevitably implies change, and that the technology always is designed and used by somebody in some social setting, it seems hard for designers to handle change as a social phenomenon. Attempts to help designers handle such phenomena have often been concentrated on how to merge or bridge the gaps between technical and socially oriented perspectives or between theory and practice. Instead our aims are to let the different perspectives talk to each other through joint activities, and provide opportunities for the participants to switch between multiple views, emphasizing even conflicting concerns. In this effort we found ourselves in activity theory.

Activity theory proposes to understand communities of practice (Lave 1988) as human activity systems, upheld by mechanisms of the (inter)action of the actors, their practice, by which they produce an outcome and reproduce themselves as competent actors. An action is understood as the purposeful human conduct through which actors relate to the object of an activity: In the beginning, when we learn to ride a bike, we are concentrating our effort on keeping the balance. Once having learned that, keeping the balance turn into an unconscious operation and we are free to think about where to ride the bike. Through our attempts to perform actions in relation to the object, sets of unconscious actions and overviews allowing for further expansion are produced. By the transformation of actions into operations and the expansion of actions into overview, complexity is reduced and the activity as a whole is expanded.

This should not be taken as if the actors act directly on the object. On the contrary, according to activity theory, all action is mediated by communication, tools and working divisions of labour, all inherent in the culture which meets the actor on a first encounter. Because activity theory is aiming to understand an activity in its historical developmental context, the theory pays much attention to the role of mediation. Mediation is taken as crystallized action, and thereby as a source from which to dig out developmental structures and transformations. The focus on mediation is more than anything else what makes activity theory so stimulating for reflections about design.

In this light an artifact is constituted through its different roles, depending of the relationships to the object and between the actors. A "role" is a label for a position under constant and mutual definition depending on these relationships. Furthermore, the artifact is the outcome of other activities, and may pass back and forth between these roles. We shall expand on this in the next section.

Design

Design is a particular kind of activity which crosses, or lives on the boundaries of, several communities of practice, relating the future to the past. Design processes have a double orientation: towards the product and towards the process. Since what is going to be created is by definition new and unknown, at least to some extent, neither product nor process can be fully known or planned in advance.

Systems development methods in general are rather detached from praxis, and they often present themselves as cook-book recipes, without concern for the specifics of the design setting or the qualifications of the designers. Furthermore such methods often prescribe a stepwise process, from an analysis of the present work situation to the programming of the system, as if the new either comes out of the blue, or is ensured by the stepwise construction process.

Designers need guidelines and plans. This is not for total prediction, but to guide the process and to get to grips with the shaping of the artifact. Thus, they need help, to assess current use, as well as to anticipate and transcend current use in a planned way and in a specific direction. *The designers need to represent and hypothesize about the computer artifact and its use* and in this endeavour they need to be supported by thinking tools. This view on design is in line with the findings of the Amodeus project, which showed how design consists of iterative processes of idea generation and evaluation (Nielsen, 1991).

When discussing what creative thinking means in the concrete, we came to recall stories about how Galileo worked, when he was developing the formula of the free fall. As far as we have been told, he used both an experimental tool and one for analytical evaluation. For experiments he used a piece of wood and a ball. For analysis of the results he used the co-ordinate system just recently invented by Descartes. What strikes us is that he used both experiments and analysis. He knew throughout that he was to look for something new. By experimenting and analysing he found a "third point" from where he could think about the free fall (the idea of a gravity point) and then suddenly experimental results and the analytical reasoning became fruitful in new ways. The virtue of this set-up is that the tools are so different and lead to different kinds of results, which can speak to each other.

The idea of "the third point", the process of formulating a hypothesis in an exploratory way, we find in what C. S. Peirce has labelled "abduction". A good abduction, according to Peirce, a. explains the facts, b. fulfils its ends, when it "through subjection to test of experiment, ... lead to the avoidance of all surprise and to the establishment of a habit of positive expectation that shall not be disappointed" (Peirce, (1934), p. 123). Peirce talks of abductions as qualified guesses. The abduction goes beyond and comes before both the deduction (where you apply a theoretical stand to a practical problem) and the induction (where solving practical problems lead you to develop a theory). Peirce's abductive, creative, or hypothesis-driven thinking, thus consists of the following: facts interpreted as traces, hypotheses of what might be the case, followed by interpretations which are attempts to match traces with potential explanations. New hypotheses following comparison with traces and old hypotheses, leads to further investigation in still new rounds. In design, these rounds have to be collective in order for the abductions/hypotheses to become shared. To support abductive thinking, both idea generation and systematic reflection must be supported for designers to efficiently capture and communicate their ideas. Once a catching design idea is born it must be subjected to test (Peirce, 1934). This is a kind of reflection that has to be carried out as a disciplined walk-through on which advantages as well as disadvantages of solutions are systematically considered. To get ideas and to reflect upon them systematically- it is the mediation of these important kinds of action in design we are going to support. At first we develop further an understanding of tools from within activity theory.

Tools

Leontjev (1978, 1981) - one of the founders of activity theory - distinguishes between three levels in human activity: a. activity - the overarching, collectively constituted integration of actions oriented towards and defined by a shared objectified motive, of which the individuals may or may not be consciously aware, b. goal-oriented individual actions of which the individual is consciously aware, can plan, discuss and modify, and c. automatic operations depending on the specific conditions in the actual setting. If we recall the example of riding a bike: when we start learning to ride a bike, biking is the activity, later it is just an operation which we unconsciously adjust to the conditions (the weather, the road etc.), or an action we can discuss as an option in line with other means of transportation. Leontjev suggested to take these levels as a hierarchy, subordinated operations to actions, and actions to activity.

Wartofsky (1979) suggests an analogous hierarchy of roles of artifacts: at the operation level, he talks about "primary artifacts". Here the role is to be transparent. If the artifact really serves us as tool, we are not consciously aware of it. At the level of action, we are consciously aware. The role of "secondary artifacts" is to preserve and transmit skills in production and use of the primary artifacts. In this way secondary artifacts become representations of the primary level. Wartofsky also suggests a tertiary role for artifacts corresponding to the activity level. The representational role characteristic for the secondary level is here suspended at the benefit of a more imaginative role which gives identity and overarching perspective to collective activity formations.

Engeström (1990) builds in his understanding of "tool-ness" on Wartofsky, and talks about the artifact playing a certain role in contextualization of actions, i.e., in answering questions of "for what", "how" and "why" artifacts are used. Accordingly he refers to the Wartofskian hierarchy in terms of 'upward' and 'downward' contextualization. The "what-role" or the role as primary artifact corresponds to the level of operations, while the "how- and why-roles" or roles as secondary artifacts serve the purpose of preserving and transmitting skills in the production and use of primary artifacts. A tool playing this role is representing the level of operations in the sense that of being symbolic externalizations or objectifications. At the tertiary level, as "imaginative" artifact, no "what", "how" and "why" question is answered, because the reference point is no longer the immediately known presence, but the activity under expansion.

Bødker (1991) has dealt with how to create artifacts that do not cause breakdowns in the fluid conduct of work. This is what Engeström (1990, p. 194) calls 'downward contextualization'. But, as Engeström points out, if the intention is to expand and transcend already known possibilities within a given work context, an 'upward' contextualization is needed as well: it has to be anticipated how the artifact-to-be will support overall conceptualization and point to new possibilities.

Engeström has labelled such expansive tools "springboards". He outlines these tertiary, imaginative artifacts in the following way: "*A springboard is a facilitative image, technique or socio-conversational constellation...misplaced or transplanted from some previous context into a new...*" (p.287). Springboards do not come about smoothly or automatically, and they are not as such solutions to the problem that one is facing. They are starters which may lead to an expansive solution. In his work with expansion of work contexts Engeström (1990) has observed an important interdependency between the roles/levels: his results seems to point out that unless a tool, meant as an imaginative artifact and instrument for expansion, is anchored also in the primary level, people tend not to use it.

In order for an artifact to serve as springboard and address the future, it also has to address the present. Addressing the present, Star & Giresemer (1989) introduce the notion of boundary objects characterizing common intellectual tools, which fulfil the role as containers and carriers: " *...both plastic enough to adapt to local needs and constraints of the several parties employing them, yet robust enough to maintain a common identity across sites. They are weakly structured in common use, and become strongly structured in individual site-use. Like a blackboard, a boundary object 'sits in the middle' of a group of actors with divergent viewpoints.*" (Star & Giresemer, 1989, p. 46).

If a tool serves as boundary object in a design situation, it represents and refers to a known use situation at the same time as it embodies the meaning assigned and taken for granted within the community of users. Following the above observation of Engeström, an artifact intended to serve as springboard must also, in collaborative settings such as design, serve as a boundary object.

This multi-functionality of artifacts is definitely a challenge to design of computer applications: not only is the artifact to - as stated by Ehn - "do something for you and remind you of something you can do" (Ehn, 1988), which in activity terms means that an artifact is mediating as well the productive as the reproductive aspects of the action - it is to do so for different people involved in the activity from different angles and with different capacities - and it has to do so in a way oriented towards the unknown, the future work situation.

If a tool serves as springboard, it represents meaning for the designers and embodies design ideas. Such embodiments do not refer to some kind of present or future use reality, for which reason they will not be of much use in stepwise refinement and modification of what is already there. Instead they may - while being created within a community of designers holding mutual perspectives and experiences - facilitate the creation of something new.

We have brought together the idea of design as an abductive way of thinking, as being in need of a third point from where to reflect, and the idea of tertiary, overarching artifacts serving as boundary objects in design. In the following we shall outline how these ideas may be implemented in CSCW design tools.

Design work & tools

Design means a focus on the process as well as the product: Focusing on the process means focus on cooperation in the process, on working division of labour, resources, etc. Focusing on the product means focusing on properties of the product (to be) and the activities that go into shaping this. The artifacts used in design carry this double determination: of engendering the decisions made in design, and of being a vehicle of communication among participants.

In systems development there is an ongoing discussion of what comes first formalisms, or exploration, where one side holds theory as the prime source of knowledge, and the other praxis. Floyd (1987) argues that both the theory driven and the situated is necessary. Where she sees the main emphasis being put on the theory driven side in most cases, our research has been more on the exploratory, empirical side. (Bødker & Grønbæk, 1991 a, and b, Bødker et al. 1991 & 1993a)

Boundary objects facilitate communication and co-operation between different, but co-operating communities of practice. There are a least two reasons why systems development tools deserve the label 'boundary': the 'container-function', and the semantics that gradually emerge, while a group of designers (and users) are working together to embody a design idea. As far as the 'container-function' is concerned, tools play a role in supporting and keeping track of the systematic reflection about the content of design ideas. Representing the mutual experience of constructing and exploring the design ideas, they make the ideas sharable and assessable for modification and critique. Emerging semantics arise during the conduct of the design activity, in the subgroups where sub-tasks are done. When participants in a larger project want to work along common lines, and according to a shared overall understanding, a boundary object may help establish a shared 'task-semantics'.

Bødker & Hammerskov (1984) showed how traditional descriptions alone are unsuitable to engender the shaping of the product, or to serve as vehicles of communication, because the proposed general semantics for this is insufficient. When it comes to communication the main problem, though, is that the shared understanding is created in the construction process, it is not inherent in the outcome (i.e. the description - see e.g. Munk-Madsen, 1978). One may say that this is the "representation-embodiment-problem" once again: the map is not the territory, and what is even more important - the territory does not necessarily exist yet.

Even though it is impossible for designers to make representations that stands for "the whole truth", some kinds of representations of the computer application being designed is necessary in order to hold on to the construction. The question is how to make such representations: To support abductive thinking, i.e. idea generation as well as systematic reflection, we have found *scenarios* as candidates: they may embody the negotiation process as well as hypotheses that to some extent embody the product to be.

The thinking toolbox needs to guide the users in reducing the empirical situations to manageable dimensions as well as in clarifying and completing the description. Lynch (1990) deals with the process of visualizing and matematisizing scientific phenomena. He points out that the stepwise process from an empirical mess to a scientific drawing is not only a matter of reducing information to manageable dimensions. Instead representations are added features to clarify, complete, extend, etc. the incomplete state of the specific, unique phenomenon represented. In this process the scientists apply generic pedagogy as well as abstract theorizing. Thus they are moving the representation towards the essential and typical, away from the specific and unique. In a similar way we want scenarios to represent the essential and typical in use situations.
In analogy to this, we have found that asking appropriate theory-driven questions about the computer application-in-use is useful. Such sets of questions we have called *checklists*.

The toolbox

Our toolbox should support shared understanding of the product and of the semantics for bringing it about, and generation, development and evaluation of hypotheses. A major part of the toolbox represents knowledge from practical design, from theoretical foundation, from inquiries and investigations, put together in forms of questions and case-like descriptions, all of which to ensure abductive thinking in the space between theory and reality.

Our work with the practical formulation of the elements of the tool box is carried out in Esprit project 6155, called EuroCODE, the aim of which is to develop a design environment for CSCW applications consisting of a shell, some demonstrators and the framework. According to the technical annex for EuroCODE, this framework should facilitate communication about the shell and the demonstrators, about the changes in goals for and quality of the outcome, emphasizing the social, educational and organizational aspects of process and product. The EuroCODE demonstrators should be prototypical examples of the kind of applications that can be built using the CSCW framework and shell. As a CSCW product, the shell must support co-ordination, communication, and sharing of materials. The framework is developed in close interplay with the development of a CSCW shell, with products and demonstrators - in collaboration between different contractors. At present we are in the middle of a second iteration of the framework. This has meanwhile been tried out by EuroCODE shell and demonstrator designers (see Bødker et al., 1993b, and 1994).

All the tools in the toolbox are meant to both speak to and contradict each other to stimulate discussion and dialogue. That is e.g. the main reason why we have chosen to make two separate checklists, one for *work* and one for *technical matters*.

Checklists and 'paradigm' cases

The technical and work-oriented checklists each are formulated under twelve headings, formulating various thought-provoking questions. These questions have been compiled from activity theory, from existing CSCW frameworks, and from the technical concerns of the EuroCODE shell and demonstrators. They are meant primarily to support examination of hypotheses and to help clarify and complete the scenarios, in the fashion described by Lynch (1990). Furthermore, to support the understanding of possible answers, and to give an outline of the current state-of-the-art, a set of key points in CSCW and CSCW-applications ("Technical solutions looking for a problem") are offered to users. These paradigm cases constitute points of departure for idea generation. We see our examples as helping the designer understand the consequences of design choices by making analogies to well-understood and clear-cut examples.

Creating scenarios

Scenarios exist in the borderland between experience and expectation, the borderland characterized by Ricoeur as being between space of experience and horizon of expectation (Ricoeur, 1988, chap. 10) and they have the power to provoke idea generation. In our approach we seek to establish this relation by offering examples of crucial and critical aspects of work which might be affected by design, and examples of possible technological solutions. Dressed up with examples and theory we assume that designers will be ready to think, i.e. to create scenarios themselves.

Scenarios are representations of the meaning that designers assign to embodiments of ideas of the future artifact and its use (see also e.g. Campbell, 1992, Carroll & Rosson, 1992, Carroll & Kellogg, 1989, Kyng 1992, in preparation). To understand more precisely what this means we will make analogies to the role of representations in scientific practice. Latour (1990) points out, using the example of Galileo as well, that exactly *development of tools for representation has been of major importance for scientific breakthroughs*, emphasizing to us the importance for the creative processes of a good toolbox.

Viewing scenarios in the role of what Wartofsky calls a tertiary or 'imaginative artifact', the role of which is 'to give identity and overarching perspective to collective activity formations' (Engeström 1990, p. 173-74), is a way to help a groups of designers ask questions like "Who has access to these files, at what level?" "Who share this material, in relation to which processes". Because scenarios are not empirical situations, they should be 'stories' located in time and space, 'traces' featuring details, not 'novels', and they should be designed based on knowledge about typical ways of doing things, but addressing specific, critical instances of the typical.

In the creation process the scenario-makers may get support from the following narrative scheme, considering *what is done* (product of activity from the point of view of the organisation and from the point of view of the different groups of involved actors), *where* (situating the activity system including artifacts), *by whom and when* (working division of labour and the order according to which the activity is carried out), *by what means* (the role of the artifacts: position in division of labour, tool functionality and communications-functionality) and *in what way* (the underlying culture, norms and values).

We will shortly summarize the qualities of the use of scenarios in design:

- they support the build-up and use of a shared understanding among the design group.
- they exist in the borderland between experience and expectation.
- they are meant to provoke new ideas.
- they constitute a theoretical anchoring of an empirical "chaos".

The scenarios as such are not physical entities. They need to be embodied to provide hands-on experiences with problems and situations. This means that we see the toolbox primarily as the professional designers' tool. The scenarios need to be explored in various games and workshops (future workshops, simulations, organisational games, dilemma games) or in prototyping, where users can get their hands on the situation.

Using the scenarios in the design process

So far, we have deliberately avoided to specify procedures of how to use the toolbox, e.g., EuroCODE designers have freely chosen any tool they felt helpful for the task at hand. Moreover, the toolbox does not propose or constrain how to combine its components with other techniques. One way of using it in interplay with other tools and methods, however, may be the following:

1. A series of scenarios runs through the design project.
2. These scenarios help span out a theory-oriented exploration of design situations, while grounded in the specific empirical setting.
3. Numerous design activities take place within the overall design activity, ranging from initial interviewing and observation to programming and testing.
4. Some of these activities, as well as the actual shaping of the scenarios are cooperative activities among users and designers, others are primarily done by professional designers.
5. Since the scenarios as such do not embody the technology, they are not available for hands-on experiences by the users. Scenarios will primarily form the basis for other activities in which they are embodied and explored. Such activities include setting the stage for and pointing at problems and solutions to be dealt with in cooperative prototyping (Bødker & Grønbæk, 1991a and b), when

using mock-ups (Ehn & Kyng, 1991), simulations, or in more systematic explorations of a running computer application for evaluation or education.

6. Furthermore, the scenarios can be designed and explored in other types of design-by-doing situations such as future workshops (Kensing & Madsen, 1991), organizational games (Ehn & Sjögren, 1991), dilemma games (Mogensen, 1993). The scenarios may be applied as well with other design approaches, e.g., data flow analysis, object-oriented analysis and design, or transaction cost analysis.

To summarize and hypothesize a little more about the use of scenarios in design, let's consider an example (fig 1).

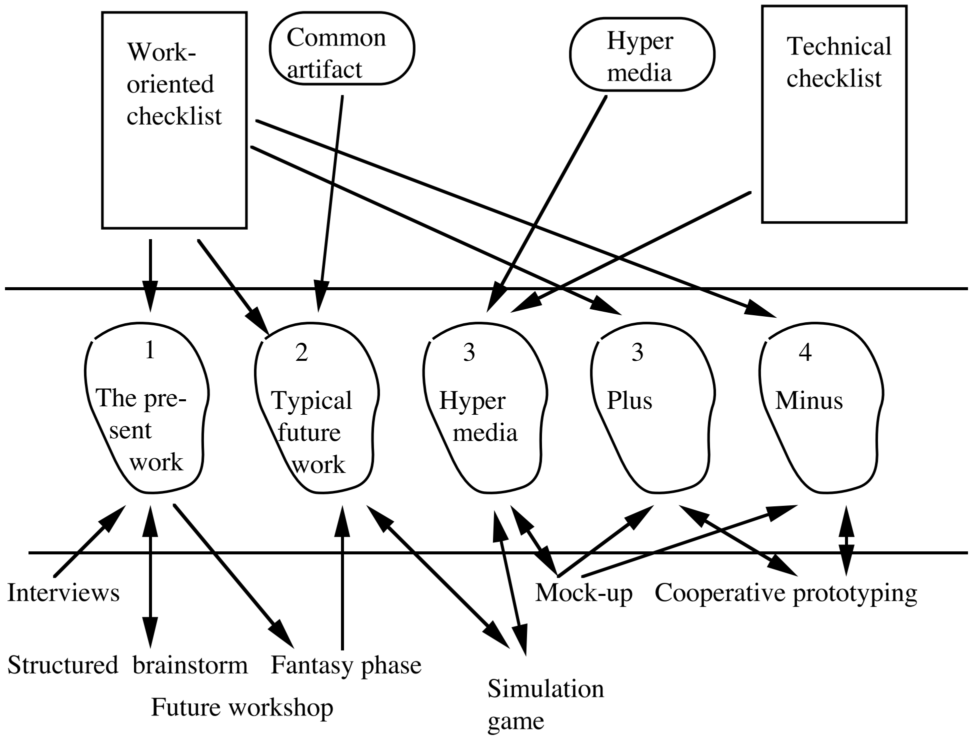

Fig. 1 "Scenario-centred" design.

Scenario 1 - the present work situation, we imagine being a scenario based on interviews and observations of work, studies of present technology, etc. The scenario may be the outcome of a future workshop, or it may be developed in an iteration with the future workshop, e.g. an early draft may set the stage for the workshop, whereas a more finished version may be produced after the workshop. The work checklist will serve to raise questions to be dealt with in the scenario. The scenario will span typical as well as critical situations of the present (potentials, problems and bottlenecks). It represents hypotheses about the problems of the current situation, thus pointing in directions for the change to be initiated. The scenario is a boundary object serving to point at a mutual understanding of problems of the current work. It will delineate and represent an area of interest for the design/change process.

Scenario 2 may be based on the fantasy phase of the future workshop. It explores primarily typical situations in the future changed work. The common artifact example is used to help generate the fantasy in an abductive way, and the work checklist to help consider the consequences of the choices made. This scenario represents a thinking from third point, and may serve as a springboard. It may be explored in a simulation game, resulting in some modifications.

Scenario 3, a hypermedia solution may be explored in close interplay with mock-ups and early prototypes based on the EuroCODE-shell and hypermedia. Initially this scenario will deal with typical situations of a technical nature, perhaps moving more and more into critical situations as the prototypes evolve. Alternative prototypes may be applied to explore these critical situations. The hypermedia example is used to enhance the technical imaginations in an abductive process, and the technical checklist to consider the potentials, problems and bottlenecks of the situations, in examining the scenarios.

Scenario 4 and 5 are plus and minus scenarios for the use of the prototypical technical solution. These scenarios represent qualified guesses about "a good" and "a bad" use situation around the activity. These solutions "talk to" the experience of users as well as designers, making it possible to discuss what is wanted and what is not. They may be explored also in other design activities such as dilemma games, and they may be used in setting the stage for cooperative prototyping activities. Work-oriented and technical checklists are used to examine the consequences, as well as to clarify and extend the representation by asking critical question about well-known consequences.

Scenarios for evaluating prototypes usually move from typical to critical as the prototypes develop vertically and horizontally (and what is typical and critical may change). As the prototyping proceeds, the process changes character from a focus on the typical work around the prototype to one where it is important to explore and get hands-on experiences with a more and more complete prototype, covering increasingly specific and peripheral work activities. This is both because it is hard to evaluate critical situations if the typical situations are not yet in place, and because the breakdowns in use spread more and more into extreme, critical situations.

Paraphrasing "Life of Brian" - What have we gained?

> "We come from nothing and we go to nothing, so what have we lost - nothing"

We should like to rephrase Monty Python's sentence saying: "We come from a scenario and we go to a scenario, so what have we gained - another scenario?

In HCI, the discussion of theory-driven versus ad-hoc design has been discussed vividly for many of the same reasons we discuss. The group around Jack Carroll has developed the use of scenarios to account for usability of the computer artifacts, and the analysis of the psychological theory "embedded" in the artifact (see e.g. Campbell, 1992, Carroll & Rosson, 1992, Carroll & Kellogg, 1989). As Bannon and Bødker (1992) point out, this is extremely problematic - without analysing it in its setting we are bound to overemphasize aspects of the artifact that may not be crucial in the use setting (See also Wixon et al., 1990). When it comes to CSCW applications, the ability to conceptualize the social setting of design and use is crucial, because these applications are dealing with social situations which are far more complicated than the one user-one computer situations, the HCI focus for many years (See e.g. discussions in Carroll, 1991).

Campbell (1992) tries to categorize scenarios, based on the assumption that a scenario refers to "representative instances of interaction between user and system". We think, much in line with Kyng (1992), that these categories are presented as a goal per se in much of the literature, rather as an instrument to be used in co-operation between users and designers, e.g. in relation to evaluation of prototypes, future workshops etc. They are all dealing with almost fully specified systems, whereas our notion is one where an early idea about a system may be explored as well. We see scenarios as a way of referring back to the user praxis, again making them no end in their own right.

Making scenarios is a creative process: they are hypotheses, or qualified guesses about the future computer application, as embodiments of it. Thus our tool-box cannot be used in a stepwise derivation of scenarios. Checklists may be used for documentation, and systematic evaluation of the design ideas, and they may be used to clarify and extend the scenarios, by pointing to directions to be covered.

Checklists, or individual items on the lists, may introduce conflicts around the computer application and its use. We have deliberately tried not to resolve such conflicts theoretically a priori, because it is in this creative space of conflicts and contradictions that something new is created (Engeström, 1987). Conflicting points of view can only be dealt with in terms of specific empirical situations. Here they may result in solutions that transcend the dilemma, or where deliberate choices are made in favour of one side of the dilemma. We find it important to work with alternative scenarios illustrating alternative paths, though we have not worked with this specific topic.

Looking at the theoretical side of design of CSCW applications we find that most frameworks give rather idealistic descriptions of what group work is (supposed to be). One exception from this is the work by Schmidt and others (Bannon & Schmidt, 1992, Schmidt, 1993). The use of this framework in design has been dealt with in detail in (Bødker & Mogensen, 1993). We have found that such a frame creates problems because it pigeonholes work as cooperative or not, instead of asking questions about *how* a certain technology makes work more, or less, cooperative. Though it claims to view cooperation as an emerging phenomenon, it describes cooperative work as something static. We view design as a process of cooperative abductive thinking in which scenarios crystallize a shared understanding of the product (to be), thus populating the space between theory-driven and situated design. This way, we have tried to avoid the ontological "trap" of discussing what CSCW is or is not. Instead we have provided a toolbox, allowing for exploration of better computer support for cooperative work in the specific work settings. This exploration is inspired and directed by theoretical concerns, as well as "ideal cases" (good or bad) which constitute state-of-the art in investigations of CSCW applications.

What we hope to gain through this effort is not just another scenario or guidelines for making better scenarios, but support for embodiments of the future product, embodiments that are open enough to allow for dialogue and sufficiently precise to provoke breakdowns in assumptions.

Acknowledgements:

The work has been funded in part by Esprit project 6155 - EuroCODE. The scenario idea owe much to discussions with Lucy Suchman and Wendy Mackay. The idea of a thinking toolbox has been developed in discussions with Morten Kyng, Kaj Grønbæk, Preben Mogensen, Kim Halskov Madsen, Peter Axel Nielsen, and Mike Robinson. The EuroCODE T1.1 group has played a major role in shaping up the idea. In particular Manfred Thüring, who has been our writing partner, has taken on the much appreciated, though little popular role of the devil's advocate. We owe much to Morten Kyng and Leigh Star for critical comments on drafts of this paper, and to Hans Roed Mark for letting us pick his brain on Peirce.

References

Bannon, L. & Schmidt, K. (1992). 'Taking CSCW Seriously. Supporting articulation work', *CSCW journal* vol. 1 nos. 1-2 (pp. 7-40).

Bannon, L. and Bødker, S. (1991). 'Beyond the Interface: Encountering Artifacts in Use'. In Carroll J. M. (Ed.) (1991). *Designing Interaction: Psychology at the Human-Computer Interface*, New York, Cambridge University Press, pp. 227-253.

Bødker, S. & Grønbæk, K. (1991a). 'Design in Action: From Prototyping by Demonstration to Cooperative Prototyping'. In J. Greenbaum, J. & M. Kyng, (eds.), *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, N.J.: Lawrence Erlbaum Associates, pp. 197-218.

Bødker, S. & Grønbæk, K. (1991b). 'Cooperative Prototyping: Users and Designers in Mutual Activity'. *International Journal of Man-Machine Studies*, Special Issue on CSCW, 34, 453-478.

Bødker, S. & Hammerskov, J. (1984). ' ISAC - A case study of systems description tools'. In Sääksjärvi, M. (ed.) *Report of the seventh Scandinavian research seminar on systemeering*, Helsinki School of Economics, pp. 201-210.

Bødker, S. & Mogensen, P. (1993). 'One woman's job is another man's articulation work - an essay about the design of computer support for cooperative work'. In Robinson, M. & Schmidt, K. (Eds.) *Developing CSCW Systems: Design Concepts*. Report of the CoTECH WG4, pp. 149-166.

Bødker, S., Christiansen, E., Ehn, P., Markussen, R., Mogensen, P. & Trigg, R. (1991). 'Computers in Context. Report from the AT project in Progress'. *Report of the 1991 NES-SAM conference*, Ebeltoft, Denmark.

Bødker, S. (1991). *Through the Interface – a Human Activity Approach to User Interface Design*, Hillsdale, NJ, Lawrence Erlbaum Associates.

Bødker, S., Christiansen, E., Ehn, P., Markussen, R., Mogensen, P., & Trigg, R. (1993a). *The AT-Project: Practical research in cooperative design* (DAIMI No. PB-454). Computer Science Department, Aarhus University, Denmark.

Bødker, S. et al. (1993b). *Deliverable D 1.1: The EuroCODE Conceptual Framework: Preliminary,* empirica, Bonn.

Bødker, S., Christiansen, E., Thüring, M. (1994) *Abductor -a framework for design of CSCW applications*, Submittet to CSCW '94

Campbell, R.L. (1992). 'Will the real scenario please stand up? *SIGCHI Bulletin*, 24 (2), 6-8.

Carroll, J. & Kellogg, W. (1989). 'Artifact as theory-nexus: Hermeneutics meets theory-based design'. In K. Bice & C. Lewis (Eds.). *Proceedings of CHI '89 conference on Human Factors in Computing Systems*, pp. 7-14, New York: ACM Press.

Carroll, J.M. & Rosson, M.B. (1992). 'Getting around the task-artifact cycle: how to make claims and design by scenario', *CACM*, 10(2), 181-210

Carroll, J. M. (Ed.) (1991). *Designing Interaction: Psychology at the Human-Computer Interface*, New York, Cambridge University Press.

Ehn, P. & Kyng, M. (1991). 'Cardboard Computers: Mocking-it-up or Hands-on the Future'. In: J. Greenbaum & M. Kyng (eds.), *Design at Work: Cooperative Design of Computer Systems*, Hillsdale, N.J.: Lawrence Erlbaum Associates, pp.169-195.

Ehn, P. & Sjögren, D. (1991). 'From System Description to Scrips for Action'. In J. Greenbaum & M. Kyng (eds.), *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, N.J.: Lawrence Erlbaum Associates, pp. 241-268.

Ehn, P. (1988). *Work-oriented design of computer artifacts*. Falköping: Arbetslivscentrum/Almqvist & Wiksell International, Hillsdale, NJ: Lawrence Erlbaum Associate.

Engeström, Y. (1987). *Learning by expanding*. Helsinki, Orienta-Konsultit.

Engeström, Y. (1990). *Learning Working and Imagining. Twelve studies in Activity Theory.* Helsinki: Orienta-Konsultit.

Floyd, C. (1987). 'Outline of a Paradigm Change in Software Engineering'. In G. Bjerknes, P. Ehn, & M. Kyng (Eds.), *Computers and democracy – a Scandinavian challenge*, (pp. 191-212). Aldershot, UK: Avebury.

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*. Hillsdale, N.J.: Lawrence Earlbaum Associates. pp. 155-168.

Greenbaum, J. & M. Kyng (Eds.) (1991). *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kyng, M. (1992). 'Scenario? Guilty!' *SIGCHI Bulletin*, 24(4), 8-9.
Kyng, M. (in preparation). 'Creating Contexts for Design'. To appear in Carrol (Ed.), *Scenario-Based Design: Envisioning Work and Technology in System Development*.

Latour, B. (1990) 'Drawing Things Together', in Lynch, M. & Woolgar, S. *Representations in Scientific Practice*, MIT Press, pp. 19-68

Lave, J (1988). *Cognition in Practice*, Cambridge University Press.

Leontjew, A. N. (1978) *Activity, Consciousness, and Personality*, Prentice-Hall (translated from Russian)

Leontjew, A. N.(1981) *Problems of the Development of the Mind*, Progress Publishers 1981 (translated from Russian)

Lynch, M. (1990) 'The Externalized Retina: Selection and matematization in the visual documentation of objects in the life sciences', in Lynch, M. & Woolgar, S. *Representations in Scientific Practice*, MIT Press, pp. 153-186

Mogensen, P. (1993). *Cooperative analysis,* Ph.D. thesis, Aarhus University.

Munk-Madsen, A. (1978) *Systembeskrivelse med brugere,* DUE-notat no. 9, University of Aarhus, (In Danish. Systems Description with Users)

Nielsen, J. (1991). Designers decision making. Design of a user interface for a picture search database. Description and analysis of an experimental workshop. Amodeus Project document: RP7/WP, The Amodeus Project, ESPRIT Basic Research Action 3066.

Peirce, C. S. (1934). *Collected papers of Charles Sanders Peirce*, vol. V. Harvard University Press, 4th print 1974.

Ricoeur, P. (1988). *Time and Narrative*, Volume 3. Chicago, The University of Chicago Press.

Schmidt, K. (1993). 'The Articulation of Cooperative Work - Requirements for Computer Support'. In Robinson, M. & Schmidt, K. (Eds.) CoTech WG4 report. *Developing CSCW Systems: Design Concepts*, pp 37-104.

Star, S.L. & Griesemer, J.R. (1989). 'Institutional Ecology,'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley’s Museum of Vertebrate Zoology', 1907-39. *Social Studies of Science* 19, 387-420.

Wartofsky, M. (1979). 'Perception, Representation and the Forms of Action: Towards a historical epistemology', in *Models*, pp. 188-210

Wixon, D, Holtzblatt, K, & Knox, S. (1990) 'Contextual design: An emergent view on systems design, in *Empowering people* CHI'90 Conference Proceedings (Chew & Whiteside, ed.), ACM, pp. 329-336.