Intertwining Training and Participatory Design for the Development of Groupware Applications

Thomas Herrmann **Kai-Uwe Loser** **Klaus Moysich**

University of Dortmund
44221 Dortmund, Germany
{herrmann;loser;moysich}@iug.cs.uni-dortmund.de

ABSTRACT
Combining participatory design of groupware systems and organizational changes with training for a modeling method and groupware applications´ usage is supposed to support the introduction of socio-technical systems in work organizations. In this paper we present our experience with the training of the employees of three companies which had started using groupware or planned to do so. From our experience in a number of designing and training sessions we have derived elements of an improved training method.

Keywords
Groupware, CSCW, Qualification

INTRODUCTION
Training and participatory design are closely interrelated. All stakeholders in the design process, especially the users, have to be well informed about the general type of system under construction and its potentials. Before people are asked to comment on a prototype, they should be able to use it and to test it. This needs training. The knowledge of the basic features of a system and their usability has not only to be conveyed shortly before the usage of a new system starts, but already for the purpose of participatory design. The ability to test a prototype is only one example amongst others, for instance, being able to analyze and to reflect tasks, to understand the documents and models of the design process etc.

Although participatory design and training are closely related, this relationship is only rarely and then not extensively discussed in participatory design literature. Kensing et al. (1996) propose a co-development of users' qualification to prepare them for using the developed system and they suggest "an initial and ongoing introduction of user representatives to the method used in the project as well as to what is expected from them …[pp 134]". Kautz (1994) emphasized the relevance of knowledge of participatory design in computing education and Mambrey et al. (1996) focus on the qualifications which are required for user advocates who have to combine knowledge of computer science with understanding the user's point of view.

This set of the relevant aspects of training described in the literature has to be enlarged if participatory design aims at systems which support communication and cooperation. This is the case with groupware where the users have not only to judge how they can control the interaction between them and the system, but also how the mediated interaction between them and others can be successfully coordinated. Thus they have to comprehend that the effect of groupware applications cannot exclusively be understood from an individual perspective.

The participatory introduction of groupware encompasses at least two aspects: the development and/or the configuration of the technical system and the structuring of the organizational system into which the technical system has to be embedded. The participants should be able to take both of these aspects into account. However, our experience reveals that also in the case of groupware, the participants are too exclusively focused on the technical aspects of the interface if their attention is not explicitly directed towards the specific needs of cooperation and communication support. Even, if the participatory design process is clearly task oriented, this does not automatically imply that the interrelation with the tasks of others is taken into account (Schwabe, 2000).

We found that participatory groupware design needs explicit training units, which have to include the following aspects:

- Learning about the specific potential of groupware
- Understanding the interrelation between organizational and technical systems and the socio-technical character of groupware
- Ability to analyze and to understand one´s own task and its relations to the tasks of others (coordination theory, Malone 1990)
- Methods to describe and to represent tasks, structures of communication and cooperation and organizational

LEAVE BLANK

relations
- Competence to comment on models of socio-technical systems
- Ability to cooperatively test a prototype

Organizing such a complex web of design, learning and usage of systems is faced with a number of problems. Combining training and participation implicates a dilemma: The less training is conducted, the poorer is the ability of the participants to understand their interests and to relate them to the system. However, the more training is offered, the more influence have the trainers who might - intentionally or not - convey their beliefs about the design process to the participants. Furthermore, groupware does not only support the usual ways of task performance but also implies that tasks are newly structured. Since the trainers are usually also the facilitators of the whole process of participation, it might easily happen that they are considered as experts for the re-structuring of the tasks. Furthermore, the complex training program can support the phenomenon of ”evolving groupware” (Orlikowski 1996) while also being distracted by this evolutionary process.

Thus, the decision of how to conduct an appropriate training in the case of participatory groupware design, how to organize the units of the training and how to choose proper methods is not trivial. The literature offers hardly any practical hints about how to find the right decisions (PDC 94,96,98). For the most part, the studies (Prinz et al., 1998, Krabbel and Wetzel, 1998) are focused on training which takes place after the customization of the system is finished. By contrast, our approach identifies the essential need for training for those units which precede this state of learning to use the finally customized system. In these units we have gathered experience of certain methods such as presenting scenarios, offering role plays, using diagram techniques to explicate organizational structures or combining prototypical screenshots with diagrams. Thus, we start with an overview of how an ideal training concept could be organized. In the following, this idealized concept is compared with our empirical experience and an extract of our essential findings. On this basis, we will conclude with a proposal of an improved training concept for the purpose of the participatory design of groupware applications.

Our empirical investigation stems from the cooperation with three different companies: The first one is a training and consulting company with about fifty employees. It offers a wide range of training courses in the field of professional further training courses. Our clients are employees whose tasks include course development, marketing, office work and administration. The second company is actually a combination of fifteen printing, media and PR companies that cooperate very closely with approx. one hundred employees. The participants of these companies are clerks and workers with a variety of different tasks. The third company is a very large one with about 6000 employees which sells adhesives, cosmetics/toiletries, detergents/household cleaners and industrial and institutional hygiene/surface technologies.

We ourselves were not only active as researchers in these different companies but also as trainers, facilitators and consultants. For the most part, it only became apparent during our training whether a member of the companies´ personnel was willing or able to play the role of an trainer, facilitator or consultant. Since we could not identify these persons in advance, we were not able to systematize our training by referring to approaches which differentiate between certain groups, such as leaders, consultants and basic participants (Timm et al. 1998) or facilitators (Yoong, 1999) or mediators (Okamura et al., 1994).

A CONCEPT OF INTERTWINING TRAINING AND PARTICIPATORY DESIGN

To handle the complexity of training requirements we focus on four learning objectives: understanding the potential of groupware applications, learning to use an appropriate modeling method, learning how to develop concepts of groupware applications in the context of organizational conventions, and learning to use groupware to carry out concrete tasks. Each sub-objective has a corresponding learning unit which can be carried out either in a one day workshop or in a number of team sessions. Although the four units are logically based on one another, we avoid strict sequencing. By contrast, the units can be partly interwoven. For example, we consider it helpful to introduce the modeling method while still being in the ”Learning groupware concepts” unit. By proceeding in this way, we can explain groupware aspects by employing simple diagrams. Figure 1 shows the four learning units and their logical sequence.

The way in which the arrows in figure 1 cut the borders of the boxes indicates that the activities can be interwoven. For instance, one can see that ”Learning the modeling method” might be started while parts of ”Learning groupware concepts” are still being carried out.

As a fundament for all units, an understanding of groupware technology and groupware concepts is necessary. The participants should get an overview of the possibilities, limitations, problems, and pitfalls of these types of multi-user applications in the context of organizational structures. The most important learning objective of this first unit is to understand that the participation in the development of groupware applications has to combine technical

and organizational aspects. Thus, the focus is to realize the relevance of organizational changes and the implications on one´s own workplace and that of others.

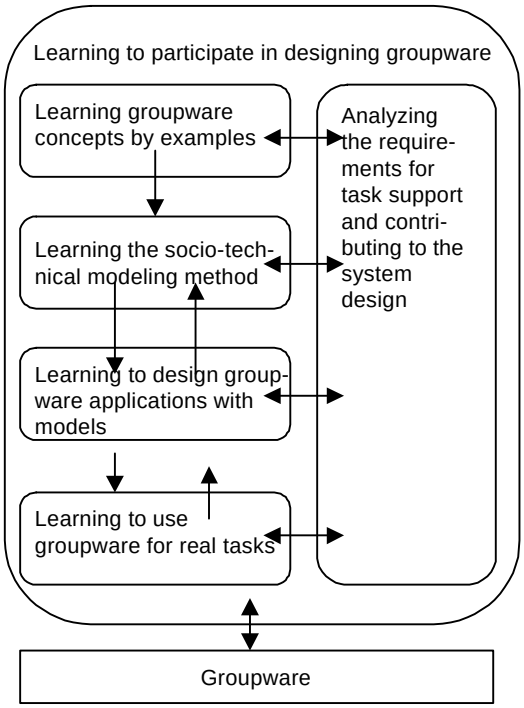

Figure 1: The learning units

To participate in designing groupware systems we find the use of an easily comprehensible modeling method essential. We have developed our own modeling method called *SeeMe*. It is a diagramming-technique for modeling semi-structured socio-technical systems (Walter and Herrmann, 1998). It provides special concepts for the representation of vagueness, incompleteness, and contradictions that are inherent to user requirements and organizational structures. Fig. 1 is an example which shows incomplete specified relations to avoid the expression of strict sequencing. More information about the specifics of and experience gathered from *SeeMe* can be found in Herrmann and Loser (1999) and Herrmann et al. (2000). SeeMe provides a third alternative to informal methods (such as rich pictures) and formal methods (such as UML or Petri Nets). We suggest the use of diagrams instead of linear text to represent complex organizational relations between roles activities, conditions and resources.

SeeMe diagrams are a means for discussing drafts of groupware systems. The learning objective of the second unit is to make the participants familiar with the basic concepts of SeeMe. They should learn to ”read” given diagrams which represent different views on the systems, such as an overview, detailed views, object oriented or activity oriented views. Furthermore, they should be able to propose modifications of the diagram according to their requirements. They should as well be able to develop basic diagrams in team work and to present them to others.

The learning of how to model a groupware application is closely related to the previous unit of learning the modeling method. While drawing and detailing examples of diagrams, more concepts and elements of SeeMe will be introduced. Aside from modeling the technical features of groupware systems, the participants should mainly learn to describe organizational conventions and social relations or implications like responsibilities, coordinative dependencies, privacy requirements, and individual interests. In joint designing sessions they should learn how to express connections and interrelations between roles, to clarify misunderstandings, to discover and discuss possible matters of conflict, and to find trade-offs in the early phases of the design process.

The fourth sub goal is learning to use concrete groupware systems. In contrast to single user applications, which are familiar to most of our clients, groupware systems require the consideration of organizational behaviour. By reflecting their expectations and experience in using groupware applications, the participants should realize the need for a critical mass of participants, and for triggers, and consequently the usefulness of organizational rules or conventions. During this learning unit, parts of other units are repeated to enable continuous improvement.

Obviously, the activities which are necessary to achieve the described sub-objectives have to be carried out cooperatively and can be supported by groupware itself. Therefore, the training units should take place in workshops and team sessions. In the following chapters we will discuss our experiences with the methods.

LEARNING GROUPWARE CONCEPTS BY EXAMPLES

In the research project LOOK[1] we taught groupware technology to the employees of two companies. The workshops with the participants of the respective companies were held separately. The first workshop for learning about groupware concepts was held with employees of the training company. In this workshop we used scenario-based training as a method. The other workshop of this unit was held with employees of a media company. In this workshop we employed role plays.

The Case of Scenario-based Training

Eleven members of staff of the training and consulting

1 http://iundg.cs.uni-dortmund.de/projekte/look/

company took part in a one day workshop where they were introduced to groupware concepts and their organizational consequences. We had some preparatory workshops, interviews and observations in advance to get an overview of the company and to gather detailed information about the employees´ working tasks. The company had not used any groupware before and offered only low bandwidth access to the internet. Only a few participants had their own email address. The location of the workshop was a special meeting room equipped with devices for video conferencing and electronic meeting support. The presenters and role players were scientists who were used to working with these systems.

We started presenting a set of typical groupware applications, e.g. video conferencing, joint editing and shared workspace, to provide an introduction to the variety of existing groupware systems. On the basis of the used systems we explained aspects of the supported communication, coordination and cooperation processes. The examples were based on a typical cooperative working task of the employees which had been derived from our earlier investigations. After the more theoretical introduction, where we used SeeMe diagrams, we presented a scenario: common tasks in preparing training courses were performed using a variety of groupware applications. A moderator explained the important steps of the stage play and directed the participants´ attention to posters and slides with corresponding SeeMe diagrams. During and after the performance we asked the employees to comment on the presentation. It was obvious that the participants had roughly comprehended the diagrams. They referred to details in the diagrams to make their contributions more expressive. This was often the case in discussions on organizational aspects.

The discussions revealed to us that the basic concepts of groupware had been understood. Previously expressed scepticism about the usefulness of groupware in their working domain had also been eradicated. All employees were motivated to participate in designing a groupware application for their company. However, we were not sure whether the participants had really realized the social and organizational aspects of groupware. One reason for our doubts was that the presented scenarios dealt with the participants´ daily tasks for which they had already developed organizational conventions. Therefore, there was no need for them to reflect on the organizational aspects consciously. To improve this learning unit in the next workshop, we replaced the scenario which had dealt with a well-known subject by a role play with a less familiar task.

The Case of Role-Play-based Training

Nine employees of the media and printing company took part in the other one-day groupware workshop. We had two sessions in advance to get an overview of the company. Like the participants of the first workshop, they had no previous experience in using groupware. The company had just started to build up an intranet and due to technical reasons only a few people had a personal email address. The workshop took place in a multimedia classroom with computers for every participant (Figure 2). We prepared the computers´ desktop according to the role plays by providing web-based clients for email, chat, team agenda, and shared workspace through a specially designed workshop portal site. Four scientists and two students, who were familiar with the groupware systems, supported the participants by answering questions and giving hints. Help pages and instructions for the role plays were also available via the browser.

Like the first workshop we started by presenting a set of typical groupware applications. This time the examples had no relation to the employees´ working field. In the following we asked the participants to carry out three supposedly simple role plays by exclusively using the provided groupware systems. For example, one of the tasks was to write down a list of all the team members´ hobbies and to publish a document with the results in the shared workspace. Using the help pages and some oral advice the participants were able to operate the groupware systems. However, they could not solve the task. In a discussion after the first role play they identified a lack of organizational rules as the main problem.It was, for example, unclear who was responsible for grouping the hobbies, and chat meetings were difficult because only few people participated and others were using a wrong chat channel. We suggested setting up some rules and responsibilities before the next task began to avoid such misunderstandings. However, they were not able to directly apply these rules. Thus, the teams could not solve the other tasks.

In the following discussion it became obvious that the employees had realized the social and organizational implications of groupware. Missing or insufficient rules were identified as the main problem. Missing time and difficulties in using the groupware systems were mentioned as the second and third important reasons for their problems in accomplishing the tasks. One participant who found the tasks to be too simple at the beginning of the workshop changed his mind later and claimed that the tasks were not closely enough related to the usual work of the company. Some found it frustrating that they could not solve a single task. All employees expressed their willingness to participate in the planned designing of a company groupware system. Some suggestions to improve the available systems, e.g. automatic notification about changes had already been made during the workshop.

Figure 2: Multimedia classroom

Comparing the two workshops, we conclude that role-play-based training is more appropriate to achieve the learning objective of this unit than scenario-based training. While the participants of the first workshop mainly discussed technical details and requirements of the presented scenario, the participants of the role-play-based workshop focused much more on the organizational requirements of groupware usage. However, the problem with role-plays can be that the users become frustrated if they cannot successfully solve the tasks.

LEARNING THE SOCIOTECHNICAL MODELING METHOD

The modeling method *SeeMe* and relations between units related to *SeeMe*

To support the design of groupware-applications we use the modeling method *SeeMe* to support the depiction of technological issues and organizational agreements. We add specific types of mock-ups – so called show-cases – to the models' abstract elements to enhance comprehension (Walter and Herrmann, 1998). The goal of teaching this method in special training sessions is to enable the users to reflect on a design which is represented in models. Furthermore, they should be able to modify these models and express new ideas using the modeling language. In order to bring modeling into practice, we applied two units: in the first unit we introduced the modeling method to larger groups to create a basic understanding in the whole organization. The teaching methods and the experiences gathered are described in this section. In the second unit we developed a model of an existing practice, the PDF-Workflow, which was recommended as an area worth discussing. Goals to be achieved, methods used and experiences gathered from this unit are presented in the following section "Learning to design groupware with models".

The goals of the initial training concepts of *SeeMe* were to create a "feeling for models", to teach abstract thinking based on diagrammatical representations of models and to teach a basic set of notational elements. The group addressed this time was the entire organization, workers who had to deal with the area of interest – the PDF-Workflow in this case – were in particular asked to participate. This group was to become especially involved in the design of further technical support for the PDF-process. The participants were urged to be able to read and understand diagrams and to be motivated into trying to express ideas of their practice using diagrams.

We had to take into consideration that the participants had varying backgrounds in their personal history and current professional position. Applied training methods could not make much use of a shared understanding of organization or practice. Neither could any of the participants be expected to have any previous experience in using diagrammatic modeling methods. Two one-day sessions were held (session A and B). The 12 participants in session A represented a great variety of position: managers, graphics designers, sales personnel, office personnel and participants of a further education program. The session took place away from the participants´ workplace. Session B took place in a seminar room at the company. The group of participants of this session was more homogeneous. Basically, two groups were participating: apprentices and the operating personnel of printing devices.

Structure and methods for teaching diagrammatic modeling

The unit was divided into three sub-phases. The first phase was supposed to create motivation to learn the method by giving a comprehensive example. The second phase was to introduce and practise the basic elements. During the third phase the participants were to do some modeling from scratch in small teams.

Motivation phase

At the beginning of the training session it is necessary to show the usefulness of modeling methods, to create motivation amongst the participants. We referred to examples and experiences of the groupware training to do so. Thus, we were referring to the participants' state of experience stemming from the initial training. The reported problems were made visible with models which

a) reflected the tasks and tools that were assigned to the problems and
b) reflected possible structures for solutions.

The first group started discussing the proposed solution presented in the diagram. They understood the solution relatively fast. Furthermore, the participants showed that they were able to follow a presentation based on a diagram

notation instantly.

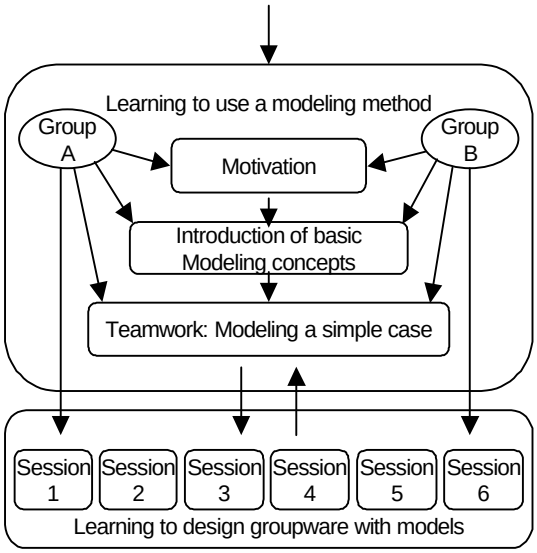

Figure 3: Structure for teaching diagrammatical modeling

The second group was more reserved. One reason could have been that most participants in that session had not participated in the Groupware training. Subsequently, the initial phase was not well oriented towards the needs of these participants. The end of this phase was based on the idea of asking for proposals to improve the presented solution depicted in the diagram.

Introduction of basic modeling concepts

In the following phase we tried to create a basic understanding of the main elements of the notation. We used a pinboard to introduce elements, present examples and do some simple practising with the participants.

Therefore, the set of notational elements was grouped into three sets. The ”basic set” consists of elements, which are absolutely necessary to draw models (e.g. roles, activities, entities and relations). The ”extension set” includes additional elements which are used more seldom or express details which are only needed in certain cases (e.g. ramifications, modifiers etc.). The ”experts´ set” includes elements which are only used in special cases and which could be omitted to reduce the complexity of the introduction. This set was excluded from the training. The basic set was introduced and taught primarily, whereas the ”extension set” was only briefly introduced by giving examples.

Exercises were then used to practise basic tasks of modeling. These tasks are needed for all modeling tasks such as commenting on diagrams or creating alternative representations. The participants learn to read models by interpreting the presented elements and how to explicate the basic structures of models: Participants were asked to assign appropriate terms to notational elements. Cases of possible ambiguity were widely discussed amongst the participants. An example might be a ”server”, which could be a role in a model of a tennis game or an entity in a model in software domains. The tutor tried to hold him back and to leave it to all the participants to find a good answer. The resulting discussions clarified the concepts. Again, the first group was participating more actively than the second. Similar tasks were used to explain ”relations between the basic elements” and the participants were asked for their ideas on possible examples. One interesting example that came up was how ”an examination” should be modeled, which is a hard task for inexperienced modelers. The discussion dealt with possible entities representing ”knowledge” and ”judgement”, the activity representing the ”task of examining” and the roles involved in this task. This led to a deeper understanding of the possibilities of the modeling notation as well as the underlying concepts.

To complete the introduction of the basic concepts we introduced a more complex example. To practise reading and reflecting diagrammatical representation, the participants were asked to ”read” the diagrams aloud. So the task was ”Explain what you see in the diagram, following its structure.”. The type of reaction which was becoming more and more noticeable was that they began with the following words: ”I don’t really understand what’s shown here, but what I see is as follows...” Obviously, they could not really understand much of the diagrams, since they were not provided with enough context. But this raised the question of the possible comprehension of diagrams without enough pre-understanding in the modeled domain. In this course it was also visible that they were able to apply the learned notation for reading and trying to understand diagrams. They tried to express some hypotheses about what could be meant by parts of the models. Interestingly, they were able to uncover some of the diagrams’ problems which resulted from inconsistencies or nonconformity with modeling rules. Because the participants’ statements also revealed that the challenges were partially too demanding, we tried to give more assistance in session B: We introduced the context of the example and presented the initial diagram that gives an overview of the successive diagrams. The task was also changed: The participants were asked to ”comprehend, explain what is depicted and discuss problems”. The example itself was left unchanged, so that the problems discovered by the first group still existed for the second group. Thus, it could be observed that the second group also discussed the major problems in the diagrams.

Teamwork: Modeling a simple case

For the final phase of the session we prepared a set of examples that were to be modeled by groups of 3 to 4 participants. The participants were prepared for this task by being given a brief introduction of how to extract facts from a

given text which was to be modeled. They were supposed to start by marking parts of sentences that might be assigned to the notational elements, a common technique for creating diagrams from texts. The participants could use paper, slides or pinboards to design the diagrams.

The resulting diagrams showed that the use of the basic set had been well understood and the groups were able to express facts using the presented notation. But the diagrams also revealed some expected problems on different levels in performing modeling tasks well:

- One group had depicted only 5 elements and their relations after the given hour. They were still discussing the directions the arrows should have, and how the facts of some phrases in the example should be interpreted. They were lost in details, instead of drawing the big picture first and concentrating on the details later. This problem is caused by the approach to work closely with a given text. We realized that the group would have benefited from more guidance from the tutors, who could have asked them to focus on the overall view first, instead of getting lost in detail.
- Another group had problems with the partitioning of a complex diagram into parts that are easier to handle. They designed a single chart with too many relations. Consequently, the drawing was not comprehensible for others. This is a known problem resulting from a bottom-up approach. Modelers need experience to be able to see possible partitions.
- Another common problem was that, in search of more guidance, they tried to copy existing examples. There was only one more complex example that had been used earlier. The overall structure of the participants' exercising task was organized similarly to this example. On one hand, this led to well structured models. On the other hand, the participants introduced elements that had never been mentioned in the texts by simply copying the patterns of the earlier example.

The models were presented to the group and the problems of the participants which occurred when designing the diagrams were discussed.

Finally, the participants reflected on what they had learned about modeling. The learned modeling methods were mainly evaluated under two aspects: the guidance for analyzing domains and the usefulness for reflecting on current work practice in principle. The two groups were very different in the overall assessment on the relevance of the modeling method for their personal day-to-day work. The first group thought of modeling as a tool that in some domains of their own work could be helpful for thinking about how work is organized. The second group thought of modeling as "being interesting in principle, but realistically not having much relevance in day-to-day practice."

To summarize the experiences gathered with this unit, we can see that users from very different backgrounds were able to work with the notation. In the diagrams the participants created, it became obvious that the groups were able to use the notation for expressing given facts. The motivation to try to express ideas from their own professional experience using diagrams differed. As explained above, the final motivation of the first group was higher, the second was lower. This difference might have been caused by the different structures of the two groups. The overall structure of the training was successful. Further improvement can be achieved by selecting more appropriate and better prepared examples. With more homogeneous groups it should be possible to select practice tasks from domains which are shared by all participants. This will make it easier to see possible relevance for personal work.

LEARNING TO DESIGN GROUPWARE WITH MODELS

With the presented training, users now should be able to understand what is depicted in diagrams and should be able to create diagrams for certain topics. What is missing then is the application of this knowledge to make proposals for groupware support for cooperative tasks in a personal context. This transfer to one´s own work context needs to be made from two directions: the modeling notation should be applied to one´s own work context and groups should learn to discuss organizational aspects using these representations. We tried to integrate both aspects in sessions where users became increasingly involved in using the modeling notation and in thinking about organizational and technological requirements resulting from their current work practice.

The method was simply to ask the participants to model a carefully selected task relevant to their work. We supported them in using the method. The activity of explaining notational constructs became less and less necessary, so that in the end we were increasingly able to behave like moderators, simply structuring the discussion.

The selected modeling task dealt with the "PDF-Workflow", the process of creating print products using the Portable Document Format technology. There were several reasons why this topic was chosen:

- For some people it is current practice in the organization.
- Some experts' knowledge is far more developed than that of many others who will soon have to work within the process. Consequently, there is a need for a knowledge transfer from these experts to future practitioners.
- There are ideas of supporting this domain with various

types of technical systems.

- The future organizational development, especially the development of new services, will have this domain as a central element.

So the goal for modeling the process was to create a foundation for training others and for discussing possible technical support for the ”PDF-Workflow”.

There were six modeling sessions of two to three hours. Six workers were regularly participating: two participants were technical experts in the domain. Two were sales personnel interacting with customers. One participant had a management position and another was a designer.

During the initial modeling session, a first diagram was created that gave an overall view of the major tasks in the process. The following two sessions were needed to fill in missing detail. The increase in explications resulting in elements in the models was enormous in these sessions. While discussing the new details, modeling constructs were introduced and applied instantly. Sometimes earlier mistakes in using the notation as well as misunderstandings between the participants were uncovered and led to corrections of the models. The fourth session dealt with peripheral tasks that are performed in parallel and are closely interrelated with the process itself. The fifth and sixth session were used to prepare a training unit for other members of the organization. The models were slightly changed to give sometimes less and other times more detail on specific parts of the process. Screenshots were integrated to give a clearer view on how the tasks are performed using the existing tools. The last three sessions operated on a model which did not increase its complexity. There were corrections, additions and deletions that produced only little difference in terms of the number of elements or similar measurements. However, the structure model gradually became more and more elaborated step-by-step. Figure 4 is the overview of the model showing the major steps of the workflow. There are altogether 18 additional diagrams describing more details that can be navigated from the overview by clicking the black areas. The reader should take the following numbers as a hint of the complexity of the final model: there are 271 basic elements (153 without repetitions) including 55 activities, 80 entities and 30 roles in the process, connected to 240 relations.

The group was able to create a very complex representation of their work practice and they discussed their knowledge in order to develop a shared understanding of this practice. The participants were able to handle the diagrams in the discussions and to create material for the training of others. In the near future we will carry out training of the PDF-Workflow for all workers in the media company based on the diagrams.

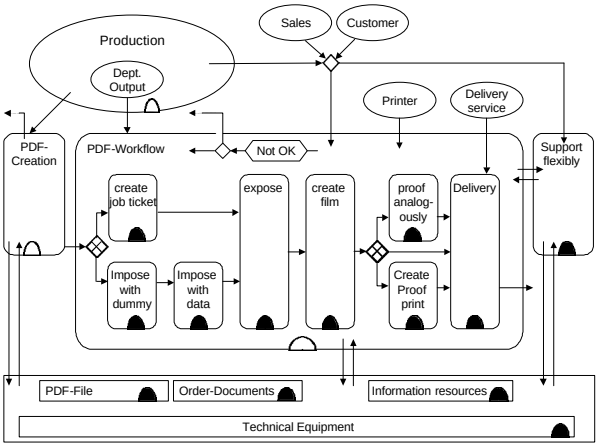

Figure 4: Overview of the model

LEARNING TO USE GROUPWARE FOR REAL TASKS

In the third case, the company had already started the roll-out of the software system. Its name is Techknowledgy® and it is a combination of a knowledge-management and a computer-based training system. The development of the system was not subject to participatory design, but it was seen to be desirable that the organizational rules for its usage and the set of content conveyed by the system be improved with the help of the users. The initial version of the system contained tutorial units for the usage of Microsoft Office. A user with a certain question can try to identify the appropriate training unit by using the system’s search engine or by browsing. The training units are linked to a series of animated screen shots dealing with a certain problem. If users are not able to solve their problem with the tutorial units they can employ the communication support features of Techknowledgy and publish their question. Thus, other users are able to read this question and try to answer it. If the other users are not able to provide a sufficient answer – for whatever reasons –, it is guaranteed that a team of experts will answer the question before 8 am the next day if it is submitted before 4 pm on the current day.

The users were expected to start with the relatively simple domain of Microsoft Office with which they were already familiar. The next step was to support the roll-out of a new Lotus Notes release with Techknowledgy without additional training courses. The overall aim was to make the users more open minded towards the cooperative features of the system and the potentials of computer-based knowledge management. Of course, the system contains information about itself.

To reach this aim, everyone who wants to become a user had to take part in a 1.5 to 2 hour presentation and discussion with about 20 to 50 other participants. Because of the large number of potential users (6000!) it was not possible

to provide training courses where they could actively use a computer or even conduct role plays. Therefore, it was decided to present a scenario to them which emphasized the cooperative characteristics of the system. The scenario was accompanied by a general introduction to the relevance and the benefits of knowledge management and an explanation of the basic features of the system. The scenario itself was based on three roles: a user who tried to solve a problem by searching a tutorial unit and – finally – by submitting a question, another user providing an answer and an expert who had to complete the answer. Furthermore, a moderator gave an introduction to the system´s functionality and how to interact with it, and she explained the activities of the scenario and commented on it.

Shortly after the presentation we conducted in-depth interviews with 16 users on different hierarchical levels and of different educational backgrounds. Since the organization of a participatory design process was only of minor relevance in this case, we focused on the question of how successful the presentation was in terms of preparing the participants for the usage and making them aware of the cooperative features. Thus, we found a set of important insights into this type of scenario-based training which helps us to improve our own training method:

- The scenario focused some of the trainees on too concrete details of the system without directing the attention towards the general possibilities and requirements – e.g. searching with the system's search-engine was only mentioned but not demonstrated as an essential part of the scenario – and therefore was not sufficiently employed by most of our interviewees.
- The general explanation of the benefits of knowledge management was hard to relate to the demonstration during the scenario.
- A lot of the interviewees were exclusively oriented towards the usage of the animated tutorial units and they were reluctant to start a communication by publishing a question in the system. They were not sure about the benefits of this possibility.
- The demonstration of the scenario was sensed as too fast for an audience which had to remain passive: "one does not know what the system is about" or "exact comprehension without practising is too difficult"
- Simple interactive input requirements were not understood. The dilemma was that a lot of details have to be known in order to operate the system, while an increase in the number of explained details complicates the understanding of the general purpose of the system
- Some people had inappropriate expectations: they expected to learn something new about MS-Office instead of learning how cooperative learning might work
- Questions which are sufficiently answered from the viewpoint of the user should be marked with a tick. This requirement was not clearly understood by everyone although it is essential and immediately clear from the viewpoint of the role of those who give answers. We assume that some participants were only taking the viewpoint of one role and that it is a real challenge to consider the possibility of changing roles.
- The trainers had differing metaphors for the system in mind which were implicitly used during the presentation such as comparing TechKnowledgy with a book, with the WorldWideWeb etc. Therefore, some of the users were confused.
- Some of the trainees were not aware of their differing kinds of needs for information and how these needs could be related to TechKnowledgy
- The trainers did not use diagrams to explain the cooperating roles and tasks. Based on the experience with the introduction of Techknowledgy, it was decided to employ diagram methods in the future to make the interrelationship between different roles more comprehensible.

CENTRAL ELEMENTS OF AN IMPROVED TRAINING METHOD

Our empirical investigations had a mainly explorative character. They gave us a number of insights which will help us to improve our method. The central aspects of improvement are summarized in this chapter. We choose to emphasize the aspects which are of high relevance for success instead of providing a complete description. *[We will apply the improved training structure in the next two month. Thus a revision of this paper could include further empirical contributions and evidence]*

Learning groupware concepts by examples

Before this unit starts, careful preparation is necessary to check the knowledge and experience which is already available. The potential of groupware should be explicitly explained, but this general presentation should be closely and interactively related to systems' presentations. The presentation should employ clear and unambiguous metaphors which will not be altered during the whole training.

Workshops with integrated role plays are the best methods for making people aware of the relevance of organizational aspects. Role plays should be alternated with phases of explanation and reflection. During the role play, help from the trainers should be available, as well as simple diagrams which explain the cooperative processing of the role-play's task. However, face-to-face communication between the participants is not allowed – they have to rely on groupware use.

The more experience the trainers have with a certain role-play the better they are able to calculate the time which is necessary or to plan the alternation between explanation, play and reflection.

A good role-play workshop offers two or three tasks for dealing with increasing complexity. The difficulties should not be caused by the content of the task but by the challenges of coordinating the usage of the technology. Therefore, the participants should not be able to apply well-known coordination conventions of their every day work life to the problem. They should be challenged to start consciously thinking about the organizational problems. Potential failures and complexity should challenge their reflection. However, the last task of the day must be such that the likelihood of the participants experiencing success is high.

We recommend the use of brainstorming methods to support reflection and to collect the items of further information which are requested by the participants at the end of the day.

Learning the socio-technical modeling method

From our experience with sessions and workshops where no diagrams were used, we derive the strong recommendation to provide diagrams of the processes of groupware use and to employ a carefully selected method. The method should be simple, extendable and it should support consistency and the presentation of organizational structures.

Success can be gained by teaching a homogeneous group with the help of a well selected set of examples which support a step-by-step introduction to the method. However, the examples should be thought-provoking and realistic to make the participants aware of the possible ambiguity of models. If prepared diagrams are used as examples, the modelers should be prepared to provide the context of these examples.

Learning to design groupware with models

The development of concepts should be related to selected tasks of the participants domain. The task should be of high relevance for communication and cooperation support. To support this – as well as the previous training phase – a set of examples (e.g. library of diagrams) which represent patterns of repeatedly occurring structures of groupware applications is helpful. The modeling method should be employed to analyze, understand and improve cooperative task performance. Optionally, this can be accompanied by developing landscapes of knowledge needed to support the tasks.

Learning to use groupware for real tasks

After a concept of the system is developed, a demonstrator, prototype or first version should be presented to explicate the basic features and to instruct the users to investigate the system cooperatively. Diagrams are useful in this training phase too. They should be displayed electronically and provide links to screen shots of the system. An overview diagram should be continuously visible. This should be supported by task-oriented scenarios and proposals for role plays. There should be push-mechanisms and triggers which support a continuous transition from the training phase to regular use.

CONCLUSION

Obviously, the proposed training concept requires a lot of work and time. For instance, during the first three training units we had 9 workshops. To add to this, we had to prepare and to continuously improve a large number of tasks for role play, diagrams and examples. However, the intensive training and its integration into the participatory design prove to be an unavoidable effort to make groupware applications a success. The training and its integration into other activities has to be planned and organized carefully. However, the way the units can be combined, alternated and intertwined is highly flexible and does not enforce strict sequencing. Therefore, the groupware application and the conventions can continuously evolve. They need not and they must not be planned in advance. This is mainly due to the close integration of usage sessions into design and intensive training from the very beginning.

We could observe that we as the trainers for the modeling method and for the different types of groupware were also considered as experts who could explain how the tasks are optimally organized and technically supported. This could be a barrier for user-driven approaches. Thus, there is further need to develop a solution which enables the trainers to reduce their influence step-by-step.

REFERENCES

1. Blomberg, J., Kensing, F. and Dykstra-Erickson, E. (eds*): PDC '96: Proceedings of the Participatory Design Conference*, 13. – 15. November 1996, Cambridge, Massachusetts, USA.

2. Chatfield, R.H., Kuhn, S. and Muller, M. (eds*): PDC '98: Proceedings of the Participatory Design Conference*, 12. – 14. November 1998, Seattle, Washington, USA.

3. Herrmann, Th. and Loser, K.-U.: *Vagueness in Models of socio-technical systems.* In: Behavior & Information Technology: Special Issue on Analysis of Cooperation and Communication 18(5), 1999, pp. 313-323

4. Herrmann, Th.; Hoffmann, M.; Loser, K-U. and Moysich, K.*: Semistructured models are surprisingly*

useful. In: Proceedings of COOP 2000.

5. Kautz, K.: *On Participatory Design and User Involvement as Topics in Computing Education: A Contribution to a Curriculum Debate.* In: Proceedings of the PDC '94, 1994, pp. 67 – 72.
6. Kensing, F., Simonsen, J. and Bødker, K.: *MUST – a method for participatory design.* In: Proceedings of the PDC '96, 1996, pp. 129 – 140.
7. Krabbel, A. and Wetzel, I.: *The Customization Process for Organizational Package Information Systems: A Challenge for Participatory Design.* In: Proceedings of the PDC '98, 1998, pp. 45 – 54.
8. Malone, T.W.: *What is Coordination Theory and How Can it help design cooperative work Systems?* In: Proceedings of the CSCW '90, 1990, pp. 357 – 370.
9. Mambrey, P., Mark, G. and Pankoke-Babatz, U.: *Integrating User Advocacy into Participatory Design: The Designers' Perspective.* In: Proceedings of the PDC '96, 1996, pp. 251 - 259.
10. Okamura, K., Fujimoto, M., Orlikowski, W. and Yates J.: *Helping CSCW Applications Succeed: The Role of Mediators in the Context of Use.* In: Proceedings of the CSCW '94, 1994, pp. 55 – 65.
11. Orlikowski, W. J.: *Improvising Organizational Transformation over Time: A Situated Change Perspective*, Information Systems Research, Vol. 7, No. 1, March 1996, pp. 63-92.
12. Prinz, W., Mark, G. and Pankoke-Babatz, U.: *Designing Groupware for Congruency in Use.* In: Proceedings of the CSCW '98, 1998, pp. 373 – 382.
13. Schwabe G. and Schenk B.: *Auf dem Weg zu einer Groupware Didaktik - Erfahrungen und Konzepte aus den Schulungen in Cuparla*. Stuttgart, 2000 (to appear).
14. Timm, E., Honecker, N., Hartmann, E.A., Gund, J. and Sell, R.: *Learning Organization – A Lasting Concept.* . In: Proceedings of the PDC '98, 1998, pp. 239 – 245.
15. Trigg, R., Anderson, S.I. and Dykstra-Erickson, E. (eds*): PDC '94: Proceedings of the Participatory Design Conference*, 27. – 28. October 1994, Chapel Hill, North Carolina, USA.
16. Walter, Th. and Herrmann, Th.: *The Relevance of Showcases for the Participative Improvement of Business Processes and Workflow-Management.* In: Proceedings of the PDC '98, 1998, pp. 117 – 127.
17. Yoong, P.: *Making sense of group support systems facilitation: a reflective practice perspective*. In: Information Technology & People, Vol. 12, No. 1, 1999, pp. 86 – 112.