

# Enabling factors for participatory design of socio-technical systems with diagrams

Kai-Uwe Loser & Thomas Herrmann

Informatics & Society

University of Dortmund, Germany

{kai-uwe.loser; thomas.herrmann}@udo.edu

## ABSTRACT

Several authors report failures when using diagrams with a defined notation for participatory design processes. Our experience in various projects was different: diagrams with graphical notations are artefacts which can be used participatively to design socio-technical systems. In this paper we describe our experience from two projects where models of socio-technical systems are designed participatively. The used methods are based on a special view on socio-technical systems. Both theory and case studies provide the basis to derive relevant factors for the process and the notation to enable participation in projects where modelling methods are used.

## Keywords

Participatory Design, Socio-technical Systems, Diagrams, Modelling

## INTRODUCTION

Diagrams in computer science are used in various fields for developing and presenting models of software systems. They are supposed to support communication for various design purposes (e.g. Harel 1988). Software systems are then created on the basis of such models. Because of this goal, diagrams focus on the technical issues (e.g. Rational 1997) and rarely present the assumptions about the organisational system behind these representations of the technical system. Some modelling methods are proposed and applied, especially in the course of business process re-engineering, to support organisational development (e.g. Scheer 1991). In our view these notations are still too formal and are not able to capture many phenomena found in organisational reality. We have developed a notation called SeeMe which proposes several concepts to represent social and technical phenomena integratedly in a single representation (Herrmann & Loser 1999).

To support the introduction of software systems as well as the organisational change which comes along with this introduction, notations need to be appropriate and appropriately used. Common practice is that a modelling technique is used by specialists, for example external

consultants. We intend to use the SeeMe diagramming technique as an instrument to support the participatory design of organisations as socio-technical systems. Ehn and his colleagues (Ehn & Sjögren 1991, Ehn 1988), for example, have already tried to apply methods like these for participatory design. They argued that one day they figured that they are the only ones to whom these descriptions made sense and therefore tried to use different methods. The fundamental - mostly theoretically motivated - critique behind this is supported by several authors e.g. Robinson & Bannon 1991. Ehn's goal was to create collective resources to help people to design their own systems by themselves. Therefore they created special representations for the organizational design. In particular, they used domain specific icons so that people can relate the representations to their own practice more easily. The artefacts are supposed to be used as collective resources for design and practice. We share these goals, but in addition we want to create a wider use of the artefacts as mediators between multiple design teams and workgroups. This is neither intended nor possible with domain specific notations.

Our approach to the problem of creating such a collective resource is to use diagrams with a defined notation. However we try to make the notation itself as well as the methods to use the notation more appropriate for participation. We use the artefacts in domains with cooperative business processes where many different viewpoints are brought together. We agree with Ehn that if only designers are involved in the creation of diagrams these are the artefacts of these designers and therefore are hardly comprehensible to others. So we try to get more people involved into the creation of the diagrams. Therefore, we have created processes where people become qualified to use the methods and to apply them for themselves, meaning that they can reflect on their own socio-technical system and make plans for the future development of their system. In consequence, the success of a project which uses diagrams should be visible by the following aspects:

- Reference to diagrams
- Participation in discussing what is, what should be and how it should be represented
- Making proposals for changes
- Changing diagrams without help

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

- Mutual learning
- Convergence of understanding
- Confidence and clarity for practice
- Confidence for changing the practice
- Plans for future use of the artefact

We applied a notation which is more appropriate for the use in socio-technical settings because it supports an integrated depiction of technical and social phenomena. A diagramming and presentation tool which supports both the collaborative processes and the creation of diagrams accompanies both methodical parts.

Many of the concepts of the method and the notation can neither be derived from case studies nor can there be any general proof of appropriateness from case studies. Only usefulness in the boundary of the individual cases can be shown. It is helpful to develop a theoretical background which grounds in the current experience and elaborate theories from relevant fields. As the notation should represent and should be used in socio-technical settings with information technology we considered certain relevant literature and theories from social sciences and general systems theory.

In this article we will derive relevant factors to enable participation in projects using such a method based on empirical studies and on theoretical insights. We start with descriptions of two case studies, where socio-technical systems are designed participatorily and continue by giving some theoretical considerations on socio-technical systems and modelling notations. In the first case study that we have already reported in PDC 2000 (Herrmann et al. 2000) we applied the diagramming notation for the first time in a complex environment and developed methods to qualify people to use such a modelling method. The focus there was the development of methods to qualify participants to develop groupware applications. During this case study hypotheses were developed about how people can make use of the method. The hypotheses we had in 2000 are more methodically explored in a study where the design for future practice was facilitated using the method. The theoretical discussions also point at a wider range of relevant factors to make such a participatory project successful. Finally, we summarize the relevant issues.

#### BASIC ELEMENTS OF SEEME

Before we describe the case studies we have to give a very brief overview of the notation SeeMe – socio-technical semi-structured modelling method – to make the examples comprehensible. A detailed description can be found in Herrmann & Loser 1999.

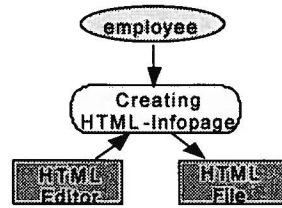


Figure 1: Simple example of a SeeMe-Diagram

**Basic Elements:** SeeMe is based on the basic concepts of role (ellipse), activity (rectangle with rounded corners), entity (rectangle) and relation (directed arcs). Roles describe a set of rights and responsibilities assigned to a person, a group or an organisational unit. Those parts of a system which can be the addressees of expectations of others (sanctions, assigning of rights etc.) are represented as roles. When persons playing a role are in action, they are performing activities. Activities use entities as resources (documents, tools, computing systems etc.) or they manipulate entities. Fig. 1 gives a simple example with the following basic elements: the role *employee*, the activity *Creating HTML-Info page* and two entities *HTML Editor* and *HTML File*. All elements in a model are at least specified by a name. More precise descriptions are possible by adding attributes or by giving detailed specifications by means of sub-elements, which are used extensively in the practical examples in this paper.

**Relations:** Relations connecting basic elements are depicted as directed arcs. Relations visualize possible logical connections as a result of a relation between two elements. All mutual combinations of basic elements are possible and have a predefined meaning. The most important examples of these meanings of relations are used in the example in Fig. 1: the role *employee* is performing the activity *creating HTML-Info page*. This task *Creating HTML Info page* uses an *HTML Editor* to create/manipulate an *HTML File*. Relations can also be qualified for other meanings.

Several concepts are added to this foundation

- to make vagueness (incompleteness and uncertainty) explicit.
- to present different kinds of attributes.
- to alter different perspectives or points of view using the same notational system.
- to integrate meta-aspects and self-reference.
- to represent social interests of roles or role playing.
- to represent but not restrict free and arbitrary decisions (contingency).

## RESULTS OF A PREVIOUS STUDY

In an earlier project – a modelling project within a printing company, describing the PDF-Workflow – a process was under investigation where learning the application and applying a modelling technique took place at the same time. There was a basic training in modelling before the modelling project started (see Herrmann et al. 2000). Learning was continued while applying the knowledge to create the diagrams for a complex cooperative task.

This study was the first participatory application of the modelling method. For smaller projects or minor tasks we already had some experience with how people work with the notation, but it had not been tried with a topic of this complexity. It was not a design project in the sense of developing something new, but a collaborative reflection on current practice.

As media for the modelling process, paper-based sketches on pin boards as well as large plots were used. This setting which was based on earlier experiences (s. Walter & Herrmann 1998) demonstrated several advantages. For the depiction of new content, the relevant elements of the graphical context were noted first on the empty wall-charts to allow the participants to orientate and set new parts in context with the already depicted areas. The experience was that ideas are quickly sketched on the wall-charts. Already modelled elements representing tasks at coarse level were used to prepare empty frames with surrounding elements for the topics of a session. Elements were depicted in this context and from there the missing detail was filled in. The existing status of the diagrams was also printed on large plots and hung around the room to be present. This made it possible for participants to refer and to make ad-hoc corrections to them. The plots have a physical presence in the room filling the walls and participants can easily focus an area under discussion. Changes to diagrams are visible as they are drawn on the plots and at the end the result of the session is visible. The problem is the correction of sketches. It is partially helpful to use cards to draw the elements, which can easily be replaced, but structural changes are still hard to visualize during the group meetings.

Although not all participants in the modelling sessions had also participated in the initial SeeMe-Training, all participants seemed to have picked up a basic understanding of the modelling technique. We think an introduction to the modelling technique is helpful, but it seemed to be possible to start a participatory design process using modelling methods almost without previous knowledge of these techniques. The learning of the techniques is done along the way, explaining the concepts with examples coming from the task at hand. This was a helpful hint, because in the participation process it is a motivational problem to introduce a modelling method without having a task that is relevant to one's own practice.

In the second case, we tried to give a very brief introduction to the modelling method only and attempted to take care to facilitate learning during the modelling sessions.

The group was able to create a very complex representation of their work practice (of course with the support of experienced modellers). Participants were not computer professionals who are used to the handling of abstract diagrams, but employees, experts in their domain, with no experience in using modelling techniques like SeeMe. With the guidance of experienced modellers they were able to express parts of their practical knowledge, referring to models and using the models and the modelling language. Help was needed to introduce new concepts, repeat explanations and give hints on similar parts of the model. It was observed that the participants referred to the diagrams in the discussions and used them.

## NEW PROCESSES FOR A NEW SOFTWARE IN A UNIVERSITY LIBRARY

### Background

At the end of 2000 we started a modelling project with two workgroups at a university library. It was planned to use a new software-system for their work on the acquisition and cataloguing of books. There is a fundamental difference between the organisational practice at that time and the type of processes required by the new technology. For this project we were asked to help the groups to redesign their new work processes using a new software system.

The traditional practice in the library was to start with the acquisition and to do the cataloguing afterwards in a sequential process. Both fields of work are carried out by specialised workgroups. Now, with the availability of software system based catalogues, the simplification of the cataloguing reduces the required qualification in most cases. What was formerly the supreme task in the library has become simple for many of the new books. There are catalogues available that already include most of the books to be acquired in the library. With the new system, cataloguing can be done in the early steps of the whole process and a high quality of data is guaranteed.

A main reason to introduce this new system is the seamless integration with the software-platforms of the other departments. However, the software system provides more functions than needed and is not sufficiently adaptable to the situation in the library. Results of the modelling project showed, that an "integrated process" will work, but that the software system creates complexity without creating enough value. Finally, the introduction of the system was cancelled, but the integrated process is now practiced as planned, using the already existing software system. For this purpose the workgroup changed the diagrams themselves. In December 2001, there was a kick-off meeting where the other members of the library were informed about the new work-process and where the new process was launched.

For the scientific analysis of the project a member of the research team created structured protocols of the modelling sessions. The last 4 sessions were also video taped. After the project we conducted semi-structured interviews and experiments with the participants to create a clearer view of what they understood and how they personally used the diagrams.

**The process of facilitating socio-technical design**

The project with the library developed a diagrammatic representation to plan and discuss what will be practised, when the system is introduced. Fig. 2 shows the overview of the diagram, including the main tasks performed during the process. At the end of the process the group should have a “shared” understanding of what the future practice will be. Therefore eleven workshops were conducted. The two leaders of the workgroups for acquisition and cataloguing who also remain practitioners, two librarians, one library employee, responsible for exploring, testing and adapting the new software, and a domain expert, also managing this project, participated in this project.

The first workshop introduced the modelling method briefly and discussed the goals for the project. The goals for the group were to design the work process, to create a documentation for reference and to exchange experiences within the workgroups including knowledge about the used software system.

In the beginning of the modelling process overviews were created showing the participating roles and the main activities of the process. Further tasks and subtasks of the main activities were informally collected to create an idea of the areas to be described. In the preliminary diagrams vagueness elements (semicircles with three dots) indicated the incompleteness of the diagrams. Participants agreed to use this diagram to organise the modelling process. During the process the participants were discussing and imagining the future practice step by step with the artefact under development. So during the following sessions the main process was filled with more detail. For the first steps this was simple, as the new system was not supposed to be used

extensively in this part of the process and much of the work was left unchanged.

For the following tasks it was not that easy: the options for different process designs were unclear and it was not possible to see what was supported by the software system and what was not. Because two participants had looked into the practice of another library, which was already using the system, the group decided to depict what they understood about the process in the other library, and then to compare this with the requirements of their library. The group was aware that the process could not just be “copied” in their own library, but that parts of the process could be reused and become part of their process. During two sessions the diagrams of the process in the other library were created. By discussing and depicting what was actual practice in the other library their own ideas became more elaborate and finally were depicted easily in a single session, reusing and referring to various parts of the diagrams that existed for the other library.

The final phase of the design process was to integrate the various special cases such as to handle norms, electronic dissertation theses, messages regarding orders, expiration of the delivery period, crediting/subsequent demand, bills, books for approval, gifts, donations or barter deals (all hidden in fig 2, visible as a residue – the black semi-circle – on *Acquisition and cataloguing*). During the final session the diagrams were checked and adapted for consistency and correctness.

After the project it was decided that the new software system would not be introduced. The discussions in the modelling project contributed some of the arguments which led to this decision because it became obvious that the system did not cater to the library’s needs, neither was it adaptable to them. However, the newly planned work practice in general was introduced. Without our help the workgroup changed the diagrams to show the use of another software system. For now, the software support remains the system that had already been in use, and the next version of the formerly planned software product will

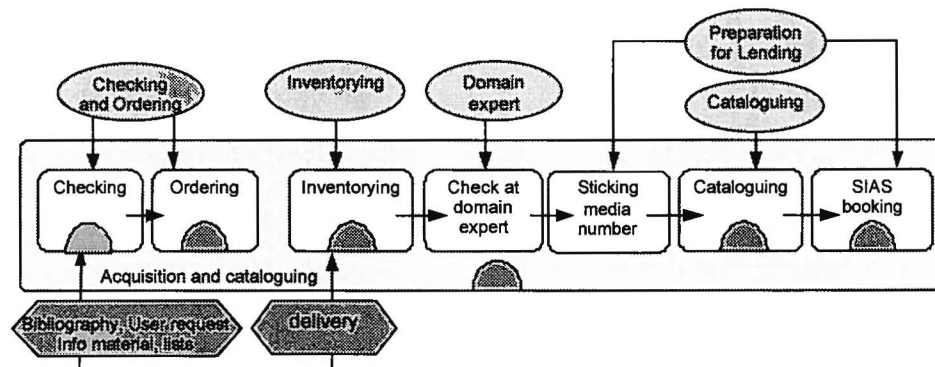


Figure 2: Overview of the acquisition and inventorying process

be evaluated. At the Kick-Off-Meeting of the new process the workgroup used the diagrams to give an overview and to explain other members of the library how the tasks will be done in the future.

### Major design issues

While developing the diagrams three areas of design were remarkable:

1. Personnel
2. Exchanging practical experience
3. Workarounds and adoption of the technical system

Personnel (1): The system's and the top management's philosophy is aiming at a "fully integrated" workplace where one employee is responsible for the whole process starting with a user's demand for a book to its integration into the library's catalogue. This guideline creates complete tasks, but it can only be partially realized with the existing personnel, because everyone should have an appropriate job. Some discussions dealt with finding and creating these jobs. Because all tasks in the process are shown in the diagrams, necessary qualifications and abilities can be evaluated. This discussion was visible in the diagrams, where the representation of a role was required although it was neither appropriate to assign a name to it nor any attributes. This role was related to a special activity which could also be carried out by other roles. However, the role had to be represented because it was necessary to give someone a job who was not able to work on more complex tasks. Since there was no sensible approach to how this role could be integrated into an official chart of the organisation, everyone agreed first on a vague solution: There is an unspecified role which puts the inventory number into the books and the question mark indicates that the correctness of the unspecified role is doubtful and therefore a rethinking will have to happen at a certain point in time. After this rethinking at the end a more complete role was set together with other more simple tasks, called *preparation for lending* (s. figure 4) that also includes the task of *SIAS booking*. Marking the role with a question mark was highlighting the role, making the speciality very obvious in the diagrams. This was to obvious for the participants, so they decided to hide this role and agreed on setting together the new role *preparation for lending*. Their argument was that both tasks are somehow related and are usually done by the person who they had had in mind.

Exchanging practical experience (2): The tasks of the employees are from the two areas of acquisition and cataloguing. Since the tasks for both are integrated at single workplaces, the exchange of experience was initiated during the modelling sessions. In both areas, things were unclear and became clearer for some participants with the discussions. Questions like: "Are you really sending a notification to every domain expert?" or "How often does this case occur?" are examples from discussions bridging

the experience gap between the two work groups. As personal understanding became visible on the walls, participants questioned their own or the visualized understanding and there was the opportunity for personal or group learning. This led to a converging understanding within the group. This was seen as one of the major outcomes of the modelling project: "Maybe,...no, not maybe, I'm quite sure, that both groups didn't know enough about each other because they had little contact. It was eye opening just how diverse the tasks are in the others domain. Before, only the necessary arrangements were known."

The interviews revealed that the result of mutual learning was one of the major positive outcomes of the project for the participants.

Workarounds and adoption of the technical system (3): As mentioned, the system did not support the specialities of this library in many situations. To overcome these deficits of the software system, the group searched for several workarounds. A simple example which is observable regularly with this kind of system is that unneeded fields in the database are used for other purposes than intended by the developers. Some of these solutions became obsolete, because the system was not to be introduced as planned earlier. The workarounds and special solutions for the system become very obvious by a comparison of the state of the diagrams at the end of the modelling project (when it was still intended to use the software) and those diagrams where these solutions are deleted from the diagrams to depict the now introduced process.

### Evaluation of the project

The development of the organisation and the adoption of the software system were closely related in this case. All of these discussions were facilitated using the diagrams as the main artefact to be created. In this sense the diagrams were used to negotiate, plan and communicate the future practice in the organisation.

In comparison to the first project the result was a design in advance rather than a reverse design and a reflection on current practice. The diagrams represent the planned practice for the groups and not the already existing practice. To imagine the future practice and discuss issues about it, presents a major challenge to such a project. There cannot be any evidence that what was planned will be in any sense a "correct" representation of what would be real practice. On the other hand all participants found it helpful, to plan and create some picture about what will happen. Basically the process led to higher practical confidence among the participants. The creation of the diagrams supported this by structuring the process and making the progress visible: „Well, I think it was an unbeatable advantage, that we had to clarify what really happens. Well, each time we became fuzzy, and meant this might be like this or like that, we weren't able to represent it or didn't want to represent it. We had to say: well let's think about what really happens at

*this specific point. I think this made it hard work and lengthened the process, but at least we have an accurate overview now."*

Discussing the role of the diagrams, it was clear that some discussions are not reflected in and with diagrams but for others, diagrams are a main focus for clarification. One example of invisible discussions is the comments on historically developed practice in the library, like the identity of the inventory number and the reference number. These are rationale that have effects on the diagrams, but they are not directly visible. An example for a discussion which was based on diagrams was discussing the way the other library was working. This comparison became structured by the state of the diagrams following the open issues which have to be depicted next. Contrasting themselves with the practice of others, it became easy to develop a model of their own practice.

It had already been discussed that the project also was initiated to provide a place for exchanging experience between the two groups. The interviews showed that the participants see this as one of the major outcomes. Creating the diagrams structured this exchange and showed the necessity for clarity about how things should be done.

During the project every participant gathered knowledge of the modelling method itself. As a result it was possible for the group to make changes to the diagrams without any help from others. They adapted the diagrams for the currently used software system on their own. Part of the interviews was a brief test of modelling knowledge, consisting of reading and discussing ability and of basic understanding of the notational elements: the average of the participants was 16 points (with a minimum of 12 and a maximum of 21). Two persons with modelling experience with the method reached a level of about 30 and two persons without any modelling experience reached approx. 8 points. In this project there was only a short structured introduction to the method at the beginning.

After using the notation in such a long project it might seem that the participants should have more knowledge about the notation. Looking at the details it is obvious that regularly used concepts are understood by most of the participants whereas concepts used less often or even only discussed and introduced are more problematic. At some point it might have been helpful to reserve a session to do some more structured qualification, to provide more details of the meanings of notational elements. The participants found a little summary page of the notation extremely helpful, it was present as poster in the early sessions, but they would have appreciated the page for the use after and before the modelling sessions.

#### **DIAGRAMS AS REPRESENTATIONS FOR SELF-REFERENTIAL SOCIO-TECHNICAL SYSTEMS**

Certain aspects of a method can be evaluated, the appropriateness of a method as a whole cannot be proved

with this kind of study. In this chapter some aspects of our method are motivated from certain accepted theories.

In both cases diagrams are used as representations of one's own socio-technical system. Usually, the term "socio-technical system" is used to emphasize that aspects of both technical as well as social sub-systems should be considered and that there is a very complex relationship between both. It is common to use this term in this simplified sense although there is a long history of the term socio-technical system. The roots are in the 1950s where the term had already been emphasized in the context of the coal mine studies of the Tavistock Institute. Later it was adopted and developed further by the Norwegian industrial democracy project. The term in this tradition is closely bound to a set of principles and values that point at a participatory approach to the introduction of technology. We agree with most arguments that lead to the development of different approaches or adaptations of this concept. But we also think that taking the original term by name and exploring the meaning in the light of new developments in systems theory and relevant social sciences can lead to helpful insights.

Included in the term socio-technical there is the distinction between social systems and technical systems. To get an idea of the difference we should think about who or what assigns something as belonging to the system or not. While living systems as well as cognitive or social systems are autonomous, technical systems are controlled from outside. Autonomy means that the behaviour of the system depends exclusively on its own structure. The behaviour is orientated towards a continuous maintaining of the system's identity and re-making of the system by itself – it is autopoietic (Maturana & Varela 1987). This process of continuous self-remaking has to be guided by a kind of description of the system which has to be integrated into it as a part of itself. This phenomenon is called self-reference (Luhmann 1993): it is the system itself which implicitly – by remaking itself – "answers" the question of which elements and relationships belong to it, or not. In the same way, the system is autonomous with respect to the question of how it reacts to influences from outside and how the perceived behaviour of its environment is transformed into information processing or into operations. By contrast, in the case of technical systems whether an element is part of the system or not is determined from outside. Technical systems serve purposes which do not lie within themselves, but are assigned from outside.

Creating a combined system with these two types we create a new phenomenon which is called a socio-technical system. Each technical system has a special relationship to the social system which produces and maintains itself but we do not consider this relationship to be the basis of a socio-technical system. The special problem is that from the very moment when a technical system becomes an

integral part of a socio-technical system, a new integrated phenomenon is developed. The high degree of integration, which makes the system a single unit, can be seen in reciprocal inscriptions (in the sense of Latour): The communication of the social sub-system reflects the characteristics of the technical system and the technical control structures mirror the properties of the social interaction.

One special feature of a socio-technical system is the way it constitutes itself. This happens with special development and adoption processes where a system incorporates new technical elements. This adoption process is a social process (e.g. Orlikowski 1996) which cannot be fully planned in advance. In our view a socio-technical system is an integrated unit. To design and develop it as such a unit, we try to support the process with external descriptions. We use graphical representations based on a given notation. Several others were proposed for this purpose, too (e.g. Ehn & Sjøgren 1991). Luhmann stated that social systems are self-referential and therefore include descriptions. This is also true for socio-technical systems; they already incorporate several types of descriptions. The description of the social-system is encoded in conventions, in the verbal elements of a shared semantic system, a meaning system or even in written rules or laws. In organizational systems coordination is a major issue that is dealt with by these descriptions. Of course technical systems are described by engineering artefacts computer systems in particular are described by software-engineering oriented modelling methods or program code. Today both types of descriptions are separated, often not made explicit and hardly reflect each other. With the modelling method SeeMe we try to bring both kinds of descriptions together and make them explicit.

In both case studies the evolution of a socio-technical system was supported with diagrams. Certain requirements for a notation arise from this. Ehn (1988) suggests a domain oriented organizational toolkit to create diagrams of the organisation which is designed for the specific domain. He states that the resulting diagrams are only used by the participants of the design project. However it is not likely that all affected people are involved during design projects. Usually the diagrams are relevant to many more people than the participants, who, where possible, should act consistently with the depicted ideas. Another point is that the group itself is not a closed social system, and has several relations to other parts of the organizational system and to the environment of the organization. The description has to support these relations and a notation is needed, which can be used in different domains.

Star's notion of boundary objects is highly relevant to this: "Boundary objects are objects that are 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." (Star 1989) A notation can try to support both requirements for boundary objects: Providing a common notation in a socio-technical system which can be used across workgroups and domains makes these descriptions robust. To create flexible descriptions, we introduced concepts to make vagueness and uncertainty explicit, leaving space for contingent action. SeeMe basically supports two concepts to describe systems vaguely:

- Intended omission of information known by a modeler (or group of modelers), but is not willing to present in a diagram.
- Expressing recognized vagueness or doubting completeness/appropriateness of contained information in a diagram.

Expressing vagueness can be used in combination with all elements: models, elements, relations, modifiers, attributes. The two basic concepts aggregate various sub-cases. One example is shown in figure 2 where the grey semi-circles show that there is detail, which has intentionally been left out in this diagram (intended omission). Grey areas (residue) can be clicked on to navigate to details that have already been modelled.

Relations can be connected to the totality of an element (specified) or to its possible parts, such as sub-elements (unspecified). Relations, at both ends, are not necessarily connected with one specific (sub-)element. The unspecified connection of a relation is especially helpful to model processes vaguely. Usually the semantics of process models is that one step is completed, then the next begins, like in state charts or petri nets. In socio-technical processes where activities can also be ongoing processes, the start of a following activity can happen at any time (vague information) while the predecessor is still active. This concept is also helpful to reduce complexity, for example, if the complete expression of all connections between two elements with many sub-elements is too complex to be shown in one diagram, the connections can be simplified to one unspecified relation or to meaningful subsets of the whole set of relations.

One example for the usage of vagueness from the library case can be found in fig. 4. There are several steps and checks considered necessary by the participants before a book should be ordered: Is it *already ordered*, is it *already existing* in the library or is there a *previous edition* a user can be referred to? After these checks certain steps follow, which are not shown in this extract. Regarding vagueness a remarkable part of the diagram shows that in some cases documents will be sent back to a domain expert. First, with the relation beginning inside the activity of *checking*, it is represented that it is not defined at which time this task starts during the performance of checking. Secondly there

is a condition showing three dots, representing that there is missing information to specify it definitely. The participants here could not or did not want to specify this condition. So this diagram is just saying: for some reason the role checking and ordering decides to send back documents to domain experts. Thus the decision is left to the people performing the tasks in a specific situation. If there was the need to specify this condition here, the participants might have chosen to neglect the task. Which might be a problem for future use like qualifying newcomers with the diagrams. The other option would be to specify the condition, which may lead to an inappropriate condition.

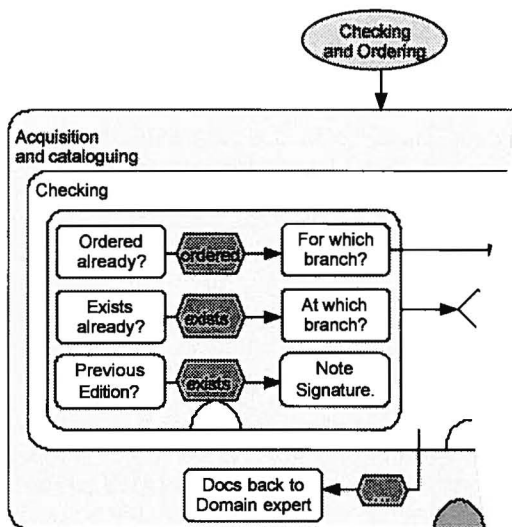


Figure 4: Example with vague Elements

Complex organizational systems create complex representations. Understanding is a major issue with complex diagrams. They are hard to follow and not everything is relevant to every recipient or participant at any time. There is a need for support for reduction of complexity. We use nesting to support this. Various types of relations between sub- and super-elements which build hierarchies are often presented using nesting of structures (e.g. Harel 1987). An example has already been mentioned: in fig. 2 the activity *Acquisition and cataloguing* is specified with its sub-activities. Embedding is used in an informal notion, simply understood as unspecified hierarchical relation. Nesting then builds a foundation for the dynamic presentation of models. Our modelling tool supports the hiding and showing of details which are embedded in elements. Additionally, not only sub-elements but also super-elements can be hidden, so that only elements on certain levels are visible. With this feature users can flexibly select the degree of visibility of the context of a certain element. The goal is to make it possible to present and develop large models that cannot be understood at once. The software tool supports the exploration of models as well as the preparation for their

presentation. We used this in various ways in the above cases. Before we had the modelling tool we used Microsoft Powerpoint to create the diagrams, only simulating the dynamics of hiding and showing with hyperlinks to defined diagrams and animations. The modelling tool now supports the hiding and showing of elements so that detailed experiments with this concept can be carried out for an evaluation.

#### FACTORS TO ENABLE PARTICIPATION

As a result from the two cases and the theoretical discussions we derive several factors which we found to have a positive effect on participation. They can be divided into two groups for the methodical parts: Modelling process and modelling notation.

##### Process factors

Regarding the process we see the following points as relevant mostly resulting from experiences made in the case studies. The first group of issues deals with the qualification and motivation of participants:

- For motivational reasons the qualification for the modelling notation should be intertwined with the task of a project group. This can be done in various ways: in a day workshop a higher level of knowledge can be reached initially like in the PDF-Example (Herrmann et al. 2000). A two to three hour introduction creates less knowledge like in the library, because there is not enough time for instant application during the session. In both approaches concepts need to be discussed in detail at the time of usage.
- It is known that such projects need a lot of personal effort and motivated participants: therefore the group should have a clear task and the power to fulfill this task autonomously. In the library project a very unclear situation was solved: the participants did not know how to deal with the task of developing an "integrated process". The problem was structured with the models so that a high motivation could be maintained during the project.

Another group of factors is concerned with the setting and technical use of the diagrams:

- So far we have had positive experience with paperbased settings. Wall charts made reference easy and simple changes could be done quickly. Sketching first drafts is easy. More complex structural changes to diagrams are not as easy, so we also want to try digital environments for this kind of project. Maybe a setting with multiple large scale displays is usable.
- In the group sessions, results of previous sessions should be visible to represent the current state of the project and to make reuse and correction possible. The rules of the notation should be present (Poster and a handout) This serves the purpose of laying out a setting



which is appropriate for the artefact and the participative process.

- The most important issue is that only minor changes are made in the absence of the participants who should develop the diagrams. Usually you would transfer diagrams between media, and adapt the visual layout for aesthetic reasons. Every simple change made in the absence of the participants needs to be presented to the participants in the following session.

One should think about which expertise should be involved in such a project. The next group of factors are related to this question:

- The facilitator needs to be an expert in moderating this kind of process. As the goal is to create a single representation, conflicts might arise. With the notation of vagueness, no single solution needs to be fixed: alternatives can be depicted as well as contingent action based on the situation. To do this, the facilitator needs to be an expert in the diagramming technique too.
- All relevant practical experience from current practice should be involved in the project.

For the modelling process itself certain steps are helpful. We have good experience with the following:

- For design projects a process should be set up as a socio-technical walkthrough. The mental process is to develop a representation, which shows how the future can be. With this process the future becomes clearer in the participants minds and the expertise of all participants is activated.
- At the beginning of a walkthrough it is helpful to create an overview. This creates an agenda which is then used to structure the process. The overview also makes the progress of a project visible.
- The modeling process should start with the representation of activities, because they can be oriented to the tasks of participants. Roles and entities are then easily completed.
- In certain situations it is helpful to search for comparable systems. Examples are processes where similar software systems are used, or discussing the software system to be introduced with other (external) users of the same system like in the library case.
- During the design the difference between what makes sense from an organizational viewpoint, and what is only required by the technical design of a software system should become clear.

#### **Factors for the notation**

Regarding the modeling notation we see the following points as relevant, most of them motivated by theoretical considerations:

- For qualification of participants it was helpful that SeeMe can be divided into subsets which can be viewed as beginners, intermediate and experts, so that you can start with simple elements and discuss more complex in the course of a project.
- To be used as boundary objects between groups in organisations, a defined notational systems is helpful. At the same time artefacts need to be flexible in use and representation. We support the flexibility of representations with concepts to show vagueness.
- To bridge boundaries and to focus group sessions representations should be reduced to relevant content. Different levels of abstraction as well as hiding and showing of certain detail should be supported.
- The results should be navigable so that participants as well as external people can view the diagrams flexibly without becoming immediately overwhelmed by the whole complexity. For similar reasons diagrams should also be presentable. This also gives some reason for supporting the creation of diagrams with a certain level of ergonomics and aesthetics.
- In the projects it was helpful that it was possible to use the notational system with different media. The first draft should easily be sketched on wallcharts, presentations should be possible (slides as well as with presentation software) and a software-based editor should be possible.

#### **CONCLUSION**

In this paper we presented two case studies where socio-technical systems were designed participatorily using diagrammatic representations. A theory of socio-technical systems re-making themselves supported by models as self-referential descriptions was outlined. Along the way we gave examples of our diagramming notation SeeMe, which makes the representation of socio-technical phenomena possible. From our experience and theoretical discussion we derived several factors which should be considered when designing a successful process.

In the two case studies people without any modelling background contributed to the design of their own practice and created a diagram that represents this practice. In the first case the application of the modelling technique helped people to get a clearer picture of their own cooperative practice and to prepare material to qualify new workers. In the second case, future practice was envisioned. Applying the modelling technique helped people to plan the necessary organisational changes in advance and to realise the problems of applying the new software system.

## REFERENCES

1. Ehn, Pelle; Sjögren, Dan (1991): >From System Descriptions to Scripts for Action. In: Greenbaum, J.; Kyng, M. (1991): *Design at Work: Cooperative Design of Computer Systems*. Hilldale, NJ: Lawrence Erlbaum.
2. Ehn, Pelle (1988): *Work-oriented Design of Computer Artifacts*. Stockholm: Arbetslivscentrum.
3. Harel, David (1988): On Visual Formalisms. In: *Communications of the ACM* Vol. 31, May 1988. pp. 514-529.
4. Herrmann, Th.; Hoffmann, M.; Kunau, G.; Loser, K.-U. (2002): *Modelling Cooperative Work: Chances and Risks of Structuring*. Accepted for Coop 2002 – Fifth Int. Conf. on the Design of Cooperative Systems.
5. Herrmann, Th.; Loser, K.-U.: (1999): Vagueness in models of socio-technical systems. In: *Behaviour and Information Technology*. Vol. 18, No.5, p. 313-323.
6. Herrmann, Th.; Loser, K.-U.; Moysich, K. (2000): Intertwining Training and Participatory Design for the Development of Groupware Applications. In: Cherkasky, T.; Greenbaum, J.; Mambrey, P.; Pors, J.K.(eds.): *Proc. of PDC 2000*, Palo Alto: CPSR. p. 106-115.
7. Luhmann, Niklas (1993 (5th ed., 1st 1987)): *Soziale Systeme. Grundriß einer allgemeinen Theorie*. Frankfurt: Suhrkamp.
8. Maturana, Humberto; Varela, Francisco (1987): *Der Baum der Erkenntnis*. Bern, München, Wien: Scherz.
9. Mumford, Enid (1987): Sociotechnical Systems Design. Evolving theory and practice. In: Bjerknes, Gro; Ehn, Pelle; Kyng, Morten (eds.) (1987): *Computers and Democracy: A Scandinavian Challenge*. Aldershot a.o.; Avebury. p. 59-77.
10. Orlikowski, Wanda J. (1996): Improvising Organizational Transformation Over Time: A Situated Change Perspective. In: *Information Systems Research*, Vol.7, No.1. p. 63-92.
11. Rational Software Corp. (Ed.) (1997): *Unified Modeling Language. Documentation Set Version 1.0*. January 1997. Santa Clara, CA: Rational Software Corp.
12. Robinson, Mike; Bannon, Liam (1991): Questioning Representations. In: Bannon, L.J.; Robinson, M.; Schmidt, K. (1991): *Proc. of the 2nd European Conf. on Computer-Supported Cooperative Work. E-CSCW '91*. Dordrecht a.o.: Kluwer Academic Publishers. p. 219-233.
13. Scheer, August-Wilhelm (1991): *Architektur integrierter Informationssysteme. Grundlagen der Unternehmensmodellierung*. Berlin: Springer.
14. Schmidt, Kjeld (1999): Of maps and scripts – the status of formal constructs in cooperative work. In: *Information and software technology 41*. Amsterdam, Elsevier. p. 319-329.
15. Star, Susan Leigh (1989): The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving. In: Huhns, M.; Gasser, L. (eds.): *Distributed Artificial Intelligence 2*. Menlo Park, CA: Morgan Kaufman. p. 37-55.
16. Walter, T.; Herrmann, Th. (1998): The Relevance of Showcases for the Participative Improvement of Business Processes and Workflow-Management. In: R. Chatfield, S. Kuhn, M. Muller (Eds.): *Proceedings of the PDC 98.*, Palo Alto: CPSR. p. 117-127.