Sociotechnical Requirements-Specification – the example of continuous support for collaborative modelling and design

Thomas Herrmann, Alexander Nolte

Department Information and Technology Management, Institute for Applied Work Science, Ruhr-University of Bochum, Germany
{thomas.herrmann, alexander.nolte}@iaw.rub.de

Abstract. The question "what is the object of socio-technical design and who is in charge of designing which aspects?" is generally described as a wicked problem which requires deliberate reflection – in theory as well as practice. Baxter's and Sommerville's differentiation between the development and change process and the development and change team provides a good basis for analysis. By reflecting on the requirements for an elicitation phase of a socio-technical solution for collaborative modelling and design in the area of business process management, it turned out that a clear separation between the technical and organizational (change-related) requirements is not possible, and therefore cannot serve as a basis for task-allocation during design. This is demonstrated by several examples of overlaps or reciprocal dependencies between both types of requirements. Furthermore, it turns out that the evaluation of a socio-technical solution additionally has to cover a variety of aspects which go beyond testing the functionality and usability of technical components.

1 Introduction

The understanding of socio-technical design implies a variety of notions depending on the questions what is – or can be – designed and who is in charge of the design:

1. The object of the design is mainly an information system which is a complex mixture of hard- and software components is either procured or developed [1]. The procurement and the development must take technical and social factors into account which must be given equal weight [21] in the course of requirements specification, systems implementation, evaluation etc.
2. The object of the design is a socio-technical system or a socio-technical process as a whole. (From the viewpoint of the authors the more appropriate focus is that on process). Consequently, socio-technical design comprises the procurement, configuration and development of technical components as well as the change of social structures and processes through managerial effort. The social dimension includes organizational aspects, value systems, people's competencies etc.

Eason [7], Cherns [3], Mumford (ETHICS [21]), Carroll (scenario-based design, [2]) and Jones and Maiden (socio-technical requirements-engineering, [16]) – understand the principles of socio-technical design according to the second notion above. They are holistically oriented and address social aspects as well as technical systems. Similarly, Baxter and Sommerville [1] point out that socio-technical systems engineering combines a systems engineering process (including procurement, analysis, construction and operation) with a change process (comprising goal setting, process mapping, process design, and process execution). However, when it comes to the question of who is in charge of which process, there is a parting of the ways:

> *"The application of this approach should feed information to the* ***development team*** *about socio-technical issues and provide support for using this information constructively in making design decisions in a timely manner. Similarly, STSE should provide the* ***change team*** *with cost-effective approaches to socio-technical analysis and provide information to them about technical factors that constrain the possibilities of change [p.11, 1, emphases by the authors]".*

Baxter and Summerville [1] thus clearly distinguish between development team and change team. The development team holds the view that their socio-technical design contributions focus on the information system, to provide a basis of sensitivity for and awareness of socio-technical issues. This focus points to the emphasis on practicality of most software projects, since any kind of responsibility of the engineers for the organizational change process would go beyond the conventional scope of these projects. Usually, system engineers are not prepared to influence the management of social or organizational change, or to be actively involved in the change process. However, the separation of development teams and change teams induce several severe problems. We experienced these problems first hand during a series of socio-technical projects, such as:

- Introducing handhelds for supporting the coordination between dispatcher and truck-drivers [13],
- Envisioning and establishing a service agency that provides services for elderly people, such as accompanying them during their weekly shopping [17],
- Establishing a system for continuous assessment of the creativity climate in organizations [22],
- Developing computer supported means for collaborative reflection at the work place [24] and
- Establishing a concept that allows for domain experts to directly influence process design through technological support [23].

From the experiences we gained throughout these projects, we conclude that the change and the development teams must at least overlap, and that we need people who are prepared to be in charge of both perspectives. This can be demonstrated as early as on the level of requirements specification: A subset of the technical requirements is basically influenced by the requirements concerning organizational change and vice versa. Furthermore, we became aware that – despite their not being strictly solution oriented – many requirements are twofold: they already include information about how an information system has to be developed from a technical perspective, as well as the description of goals to be pursued by organizational change. Before we demonstrate these overlapping phenomena, we will outline a further differentiation with respect to the object of socio-technical design.

From the viewpoint of engineering and design it seems to be advantageous to employ the concept of "system" which is to be designed: there are elements, components and relations among them which can be identified. Even more important: a system is considered to be a clearly definable unit which has clear boundaries against its environment [4]. While it seems to be relatively obvious how a technical system can be described and identified, it is much more complex to apply this system perspective to "social system". The sociologist Luhmann [18] has decisively contributed to an understanding of social phenomena from the viewpoint of system theory. He emphasizes the relevance of communicative interactions as constituents of social systems. Roughly speaking, the characteristics of a social system, its identity and its boundaries, depend on these interactions. On the one hand, a social system is open, as it interacts with its environments. On the other hand, the way it perceives and reacts to changes in its environment cannot be determined from the outside but depends on the social system itself. Consequently, social systems cannot be engineered but only be systematically influenced. Continuous change is a constitutive characteristic of social systems and therefore also of socio-technical systems. This is reflected in Floyd et al.´s [11] concept for evolutionary software development, or in Fischer and Oswald's [10] concept of evolutionary growth (in the context of seeding and re-seeding). Furthermore, Fischer and Herrmann [9] propose socio-technical meta-design as an approach for dealing with the need to react to ongoing change.

One way to influence a socio-technical system's evolution is to design plans for how the roles within this system should behave and can be supported to behave. Such a behavior-oriented plan describes more a socio-technical process than a system. The process intertwines three types of interactions:

1. social interaction – mainly communication – between people or within and between organizational units
2. human–computer interaction
3. data exchange between technical components

For the purpose of socio-technical design it is reasonable to document these three types of interactions and the way they are sequenced and iteratively intertwined with diagrams which make possible an holistic view and specification of system requirements from both system development and management of change perspectives [14].

In the following sections, we refer to a socio-technical solution (WASCoMo, *Web-based Annotation System for Collaborative Modeling)* which we have developed ourselves and evaluated in a series of concrete cases (c.f. section 2). The problem solved by this solution belongs to the field of business process management (BPM). There domain experts should continuously be involved in the re-design of business processes but are not able to do so, since they do not have enough knowledge about modelling methods. To overcome this deficit, we have developed a mixture of organizational measures and technical features. We started by developing separate requirements with respect to organizational and technical issues. Reflecting on this strategy from a distance reveals that this separation does not work successfully. We derive some lessons learnt from this experience with respect to socio-technical requirements engineering and the methods for evaluation which are related to these requirements. These lessons go beyond other approaches of socio-technical requirements engineering (cf. [8, 20]) which – in accordance with Case 1 described at the beginning of this introduction – focus on deliberately exploring the social context to improve the quality of the information system being designed.

The remainder of the paper is structured as follows: After describing the case which was the basis for the development of WASCoMo (c.f. section 2), we separately look at the challenges we faced with respect to requirements specification (c.f. section 3) and evaluation (c.f. section 4). The paper concludes with implications that we have drawn from our experiences, as well as suggestions for future work (c.f. section 5).

2 Developing WASCoMo as a socio-technical solution – an example

This experience report focusses on one of several projects that were the basis for the development of WASCoMo. The project took place in the administration of a large German university. There the department for maintenance and construction of buildings was faced with the task of (re-)designing their work processes, due to some major changes within the organization as well as due to changing external requirements. These changes not only affected single units within that organization but rather the whole department and even other departments, such as accounting.

The main problem when starting a project like this is that an initial knowledge gap between process analysts and stakeholders has to be overcome. Process analysts often do not come from within the organization – which also was the case here – and are thus not knowledgeable about the processes that have to be analyzed and (re-)designed. There are a number of ways to overcome this knowledge gap, such as through interviews, observations, document analysis and ethnographic approaches (c.f. [6] for a detailed overview). However, in order to understand a process in depth, it is often necessary to provide a means of discourse, not only between stakeholders and process analysts. but also among stakeholders directly. In order to foster this discourse, we did not rely on interviewing stakeholders. Instead, we organized workshops where, supported by process analysts, the stakeholders discussed a process and visualized it using a modeling notation [14, 25]. It took multiple workshops to arrive at reasonable solutions for altering a process or creating a new one. However, during these workshops we realized that they – due to their mode of collaboration – had a number of weaknesses. One of these is known as production blocking [5] in cognitive psychology. Production blocking occurs when participants have to listen while others speak and therefore are hindered in developing their own flow of ideas. They also might forget parts of their own ideas while waiting for their turn. Another phenomenon that might occur is cognitive inertia [15]. It means that people get stuck within certain boundaries of thinking, which might prove to be disadvantageous especially when processes have to be (re-)designed.

Furthermore, it appeared to be a disadvantage that the workshops were separated from the work context under discussion. This separation resulted in people forgetting important details. Workshops also only allow for involving a small number of participants, and it is hard to arrange workshops on demand. Especially work context relations became critical during the project. It happened often that the participants arrived at a point during the discussion which could not be solved, either because the people that had the relevant knowledge were not present or decisions had to be postponed due to issues of respect for authority or competence.

It thus seemed reasonable to us to complement these workshops with other modes of collaboration that allow for stakeholders to continuously access and review models. However, this left us with the problem that stakeholders are usually not trained in using process models. Our idea about how to solve this issue stemmed from a project in which we tried to support continuous improvement in a factory context by visualizing processes and using them as a basis for discussing incremental improvements. After we had drafted an initial model of a process, we printed it on a large piece of paper, put it on a wall and discussed how to improve the process with people who were involved in it (metal workers and superiors). During the discussion, we gave the participants the opportunity to directly alter the model by writing on it with a felt pen. The result of such a discussion is visualized in Fig. 1[1].

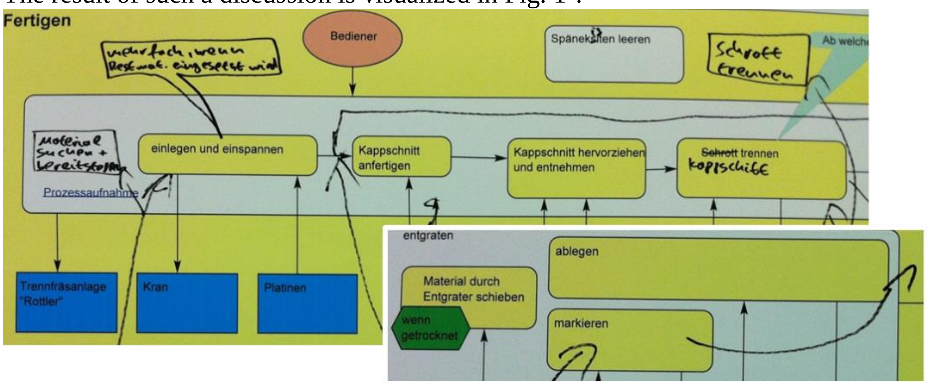

Fig. 1. Process stakeholders commenting on models

The writing as well as our observations indicate that the participants quickly grasped the idea of using the model as a means of discussing the process as it currently is and identifying ways to improve it. Our observations of people who were not familiar with process models using them successfully in this way gave us the idea that it might be feasible to provide them with a simplified modeling notation (e.g. SeeMe, [12][2]) combining two modes of collaborative modelling:

[1] Please note that the labels of the elements as well as the comments are in German as the workshop was conducted in a German factory.

[2] see http://seeme-imtm.de for more details

1. Workshops where the stakeholders are supported by experts with respect to modeling and by facilitators (c.f. initial modeling phase and model refinement phase in Fig. 2) and
2. Phases of asynchronous inspection of business process models which are accessed via a web-browser, and a tool enabling the annotation of comments – or even threads of comments – to elements of process models (c.f. annotation phase in Fig. 2).

Supporting an approach like this inevitably required a socio-technical solution. There had to be a technical component that allowed people to access a model via a web browser and create annotations on it. However, just providing such a system was not sufficient to encourage consistent participation. There had to be an organizational underpinning that not only took the transition between workshops and asynchronous phases into account. We also had to conduct workshops in a specific way that would lead to useful assignments during the annotation phase and acknowledgement of the annotations once they had been made.

We thus came up with a socio-technical solution that covered three phases (c.f. Fig. 2 for an overview). The first phase was to analyze and (re-)design the process of handling requests by students and employees of the university. Most of these requests had to do with malfunctions in the university´s infrastructure, such as dysfunctional doors. During this first phase (c.f. initial modeling phase in Fig. 2) – which can be viewed as a preparation for the other two phases – we created an initial draft of a process model covering the request handling process, from when a request is filed until it has reached the responsible unit, e.g. the unit responsible for fixing the broken door. This first draft was created during a workshop in which employees of several units of the construction department, employees of other university branches and other stakeholders were supported by modeling experts in discussing the process and visualizing it as a process model. This model then served as a basis for the following second phase (c.f. annotation phase in Fig. 2) where the workshop participants were given access to the model via a web-based modeling tool. During this phase they asynchronously inspected the model, created comments on it and discussed the process that was depicted in the model. Afterwards, in the third phase, the annotated model was jointly inspected and refined based upon the comments that had been created during the previous phase (c.f. model refinement phase in Fig. 2). This refinement phase took place during a workshop in which the same stakeholders were again supported by modeling experts in discussing the comments and altering the process model accordingly. While we only conducted two workshops in this project,, the annotation and refinement phases in principle could be repeated until the goal is reached that was originally stated when process analysis started, and until the needs of the participating stakeholders have completely been taken into account.

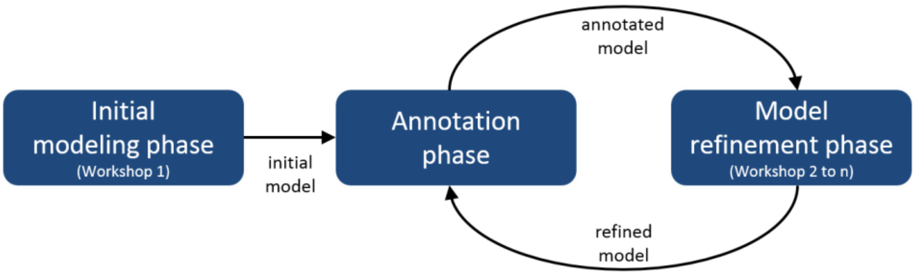

Fig. 2. Phases of the WASCoMo concept

Designing WASCoMo thus required a socio-technical concept which integrated the points of view of technical development and organizational change. This concept includes (a) the requirements specification and (b) the evaluation procedure which leads to cyclic improvement. Furthermore, WASCoMo supports socio-technical design itself, as it allows stakeholders to continuously take part in designing socio-technical processes. WASCoMo thus supports socio-technical design while being the result of a socio-technical design process itself.

In what follows we will focus on designing WASCoMo as a socio-technical design project. We will take both requirements specification (c.f. section 3) and the evaluation procedure (c.f. section 4) for WASCoMo into account. We will show how the perspectives of social interaction, human–computer interaction and data exchange have to be intertwined when conducting a socio-technical design project.

3 Specifying requirements for a socio-technical concept

Specifying requirements for WASCoMo was conducted in a cyclic process (c.f. Fig. 3). We started out by reviewing relevant literature (c.f. Fig. 3 top) and identifying some initial requirements. Parallel to this, we conducted a number of small empirical studies in which we observed stakeholders working with process models, and we also analyzed existing web-based modeling tools. Taking all these resources into account, we specified requirements that were then used to develop the organizational concept as well as the technical support underlying WASCoMo (c.f. green bubbles in Fig. 3). However, adjusting the organizational concept and the technical support to each other was part of the cyclic improvement process. This adjustment sometimes even led to the necessity of specifying more requirements, thus going back to literature analysis or observations.

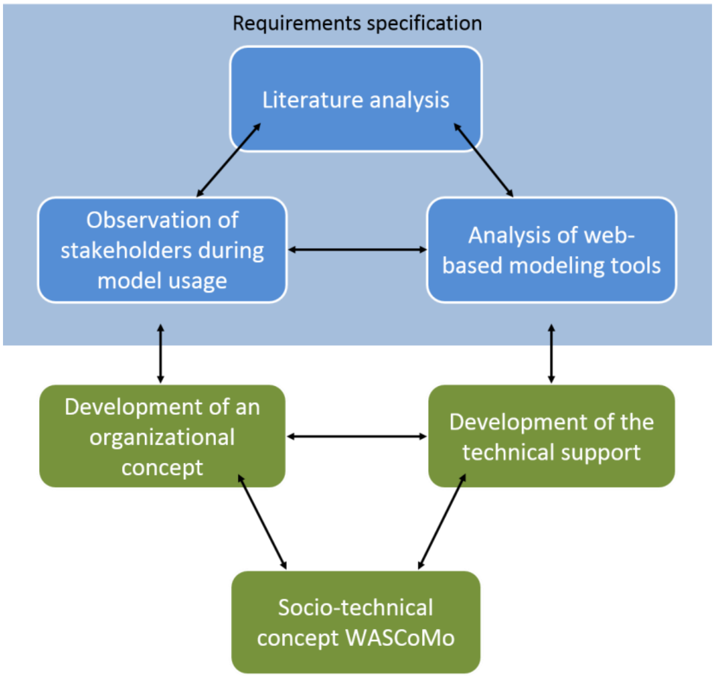

Fig. 3. Requirements specification for WASCoMo

In this section, we will focus on the requirements that we identified during requirements specification (c.f. top half in Fig. 3), thus leaving the formative evaluation to the following section (c.f. section 4).

There are examples of requirements which are clearly **technically focused** such as:

The most recent version of the respective model has to be accessible for all participants. This includes the process model as well as the annotations that have been created up to that point.

This requirement addresses the need for data exchange between the continuing versions of a process diagram at the users‘ web browsers and the server. Together with the following requirements it is obvious that there is already an expectation of what the interaction among users might look like:

The system has to support the development of communication threads which are based on the annotations being added to the elements or relations within a process model.

These requirements present the possibility that the users’ communication could be directly mediated by the WASCoMo web-application. Such a possibility raises the question whether – from the viewpoint of an organization – working collaboratively with a process diagram is an option or whether it is mandatory which would in turn bring up the possibility for the need of an organizational requirement. Instead of working like a synchronous communication channel, WASCoMo could alternatively pursue a “first approve then release” paradigm. This is an escalation scheme in which changes are firstly exclusively displayed to the originator, then to a sub-group and only afterwards published throughout an organization. This consideration leads us to the following insight: If we assume that an organization is stable with a fixed communication culture, the example of the requirement above remains mainly technically oriented. As soon as it is expected that the new technology will imply organizational change, nearly every technical requirement has to be accompanied by an additional organizational or social requirement to deal with the social implications.

Other requirements definitely address **human-computer interaction,** such as:

The positioning of an annotation within a model has to serve as an indicator for a reference between a specific part of a model and the subsequent? annotations.

However, there are requirements with a clear usability focus, such as

Annotations that have been added by others since a user visited the model for the last time have to be highlighted.

But actually they are also related to the need for increased awareness, which was identified throughout the formative evaluation as a decisive principle to encourage more participation. Consequently, the intention of this requirement has to be seen in the context of other requirements which point to organizational issues: People are expected to realize that annotations of others might be of relevance for their perspective on the business process under discussion. Again, this is a clear example that requirement engineers have to be aware of whether a technical requirement is associated with an expectation of organizational change in the system’s context. If yes, those organizationally oriented requirements have to be made explicit and the relationship between them and technical specifications has to be documented.

Other requirements such as

Participants have to have access to the models directly at the work place.

state the need for **technical as well as for organizational measures**. This is an example of the fact that within socio-technical design, the differentiation between technical and organizational requirements is easily blurred. From this example we can see that it is not feasible in every case simply to formulate a pair of requirements – one on the technical the other on the organizational level. Such a twofold specification would imply a certain solution for the problem which is addressed by the requirement. Although the requirement was meant to be considered for the web-application under development, it could also have been fulfilled by organizational means with technology that was already available in the system’s context.

Typical examples of **organizational requirements** are:

The meaning of elements and relations between elements within a process model has to be explained to the participants.

Participants have to be selected based upon their expertise with respect to the process in question, the domain in which the process is conducted, their interest in analyzing and (re-)designing the process, their openness with respect to change and their role and their standing within the organization.

The first requirement addresses the competences which have to be developed among the users. The second requirement addresses the task description of certain roles involved: somebody has to develop the criteria for identifying appropriate participants; somebody else has to apply these criteria. At first glance these requirements appear to be relatively independent from technical considerations. However, the example of selecting participants is implicitly influenced by an assumption about how many participants will be able to take part and whether they are willing to invest their time not only during workshops but also between them, and to be aware of what others are doing during this time. All in all it is about finding people who fit with the possibilities and expectations assigned to the system under development. Consequently, a requirement on the social side has to be checked to see whether and how the measures to be taken to fulfill it comply with technical features of the new system.

Furthermore, organizational issues will entail a certain degree of vagueness:

Stakeholders and modeling experts have to agree on a suitable time span for the annotation phase.

The project manager or the facilitator have to decide which time span might be *suitable*. Furthermore, it should be taken into consideration whether data will be technically provided to support this decision and whether technical means will be employed to ensure that the participants comply with the time span. It turns out to be a necessity to clarify whether each kind of vagueness associated with social requirements is intentional or whether it is undesirable. In the first case one has to accept maintaining the vagueness while designing a technical system, while in the second case the system should contribute to making organizational procedures more precise and unambiguous.

Obviously, the differences between technical and organizational requirements depends on the way they are put into practice. It seems more appropriate to firstly avoid a difference between organizational and technical requirements, since this is a principle of requirements engineering. However, it remains unclear how the division of work and the task allocation between the development team and the change team should work. Consequently, a definite overlap of these teams and their reciprocal responsibilities is needed as well as close collaboration.

4 Challenges of socio-technical evaluation

The formative evaluation of WASCoMo proved to be equally as challenging as the requirements specification above. It was not sufficient only to test the system's functionality against the technical requirements, as they had originated from the social context in which the system was embedded. It was also not sufficient to solely focus on the system's usability. Instead the socio-technical process as a whole had to be tested. We will again provide examples for this assumption.

When evaluating WASCoMo we used a number of different approaches. These approaches took the artifacts resulting from the project into account, the quality of the technical support and the quality of the collaboration as it was perceived by the participants. For the latter we conducted a short survey after each workshop (c.f. initial modeling phase and model refinement phase in Fig. 2). The survey included questions that, for example, aimed at identifying whether or not a person felt that her perspective had sufficiently been considered during the discussion. We also conducted semi-structured interviews with selected participants in order to gain a deeper understanding about how they perceived the WASCoMo approach as a whole as well as its outcome.

The survey also included questions about the resulting artifacts, such as whether or not all relevant details of a process were included into the model. With respect to the quality of the resulting artifacts we also used measures for model complexity in order to find evidence for whether process models had become too complicated to understand. We also developed a rating scale for model quality that covered aspects such as the number of modifications that had been made during a workshop based upon an annotation, or the number of replies an annotation had received which served as a measure for collaboration. We also asked participants about the usability of the web-based system and about their usage strategies. The latter was complemented with a server log analysis that provided a record for when people used the system, how often and for how long.

In the following, we will focus on interview statements, as they provide in-depth insights into how the users perceived the WASCoMo concept as a whole.

From the interviews we saw that the effectiveness of a socio-technical concept such as WASCoMo strongly depends on motivational factors and users' interest. This became obvious by interview statements such as:

> *"I don't have a strong interest in creating a very detailed model as I will not be the one using it afterwards"*

Further analysis of the server logs and the questionnaires showed that at least a third of all participants did not even look at the model during the annotation phase. The participants, however, ranked the usability of the system to be almost flawless. Considering this result leads us to the conclusion that evaluating a technical system out of context is not sufficient, as in this case the relatively low level of participation was not caused by perceived usability deficits. More probably low participation stemmed from an insufficient effort to motivate people to use or even try the technical system. This in turn means that testing technical features alone does not provide sufficient evidence for whether or not a solution to a problem will be successful, i.e. perceived as useful, and used in an effective way. Rather, in order to evaluate a socio-technical approach it is necessary to also take the quality of the organizational measures into account which are related to the question whether technical requirements are fulfilled.

Consequently, it has to be noted that the perceived quality of the outcome of a socio-technical concept is also not solely related to the functionality of the technical part. The perceived quality depends instead on how users perceive the interplay between social interaction and human-computer interaction. Evaluation results from WASCoMo provide support for this assumption. For example, one user stated that:

> *"I have talked with [participant1] and [participant2] about the model […] I integrated the result of our communication into the model afterwards"*

The perceived quality thus depends on more than a system that provides the possibility to use annotations on models as a means of communication. The perceived quality also depends on the social interaction between users, with the technical system only serving as a tool for documenting communication results. This indicates again that the evaluation of the human-computer interaction –e.g. with respect to usability– always has to be seen in the context of the accompanying communication between the users.

Furthermore, the project revealed the necessity to understand and to take into account the participants' competencies. As outlined above, the problem to be solved was derived from the low state of knowledge about business process modeling among the domain experts. Consequently, the status of this competence has to be covered by the evaluation, as well as the efforts to influence knowledge about modelling and the learning progress, both of which are key parts of participating in the workshops. We found that about 78% of our participants could be considered lay persons with respect to modelling, while the rest had profound experience with process modelling. With respect to this differentiation it turned out that experienced modelers focused on the process as a whole while lay modelers focused more smaller details. The concluding insight of this result is that the evaluation of a socio-technical process has to take the initial competencies and the learning progress of the participants into account.

Thus, the evaluation of a socio-technical concept requires a holistic approach. Although testing technical components without a social context might provide some evidence – e.g. for missing features – it is not sufficient for evaluating a socio-technical concept. Testing individual components is not feasible, as users are aware of the socio-technical environment and act within it as a whole. They do not distinguish between technical support and organizational changes. Consequently, even when a piece of software can be considered usable, the users still might perceive it as being useless. For instance, users might come to that judgment because a system does not fit their style of work.

Furthermore, it is well known that the evaluation of a socio-technical process has to be carried out through a series of usage periods. As mentioned in the introduction, socio-technical processes are subject to continuous change and evolution, depending on their context and on their own dynamics. This is especially relevant during the transition from the phase of introduction to a more routinized usage. During this transition, people influence each other's behavior and new conventions [19] emerge which have to be taken into account by the evaluation. One example that provides an indication for that necessity is the following requirement:

Participants that are more active have to motivate other participants to create annotations e.g. by asking them questions.

In order to support participants to communicatively encourage each other, technical support clearly is required. However, in order for this requirement to be fulfilled, the evaluation also has to address the hints for organizational change which should prompt participants to answer questions they receive.

5 Conclusion and outlook

While it would be feasible from a practical point of view to clearly distinguish between organizational and technical requirements in socio-technical projects, this separation is not entirely possible. We demonstrated this by analyzing multiple examples from the socio-technical project WASCoMo. Referring back to Baxter and Sommerville [1] a clear differentiation between tasks that should be carried out either by the development team or by the change team is thus not possible. Such a clear differentiation can neither be maintained in the requirements elicitation nor in the evaluation of a socio-technical process. We have provided examples for both.

We gained the following practical insights from our experience with the project described above: When requirements are derived from problems which should be dealt with by a socio-technical design, one should first try to draft them neutrally according to whether they are technically or socially oriented. Afterwards it can be decided which mix of requirements for either organizational measures or technical features fits best to meet the original socio-technical requirement. If this neutrality is not possible, or as soon as the further refinement of requirements imposes a differentiation between organizational / social vs. technical concerns, for each technical requirement it should be considered whether:

1. It has to be complemented by a new organizational requirement, so that its purpose can be fulfilled.
2. It influences the effectiveness of other organizational requirements and whether this influence has to lead to an adaptation of already stated requirement descriptions.
3. There is a reciprocal interdependency with other organizational requirements which have to be taken into account.

These considerations apply vice versa to organizational requirements and their relationship to technical aspects. All in all, the reciprocal relations have to be documented to support the implementation of a solution and its evaluation. For the cyclic, formative evaluation it is highly recommended always to check the socio-technical solution against a combination of social and technical requirements instead of analyzing them separately. Especially human-computer interaction has to be analyzed in the context of the communication between the users. The way organizational measures are implemented and how they influence the development of the users' competencies is a crucial aspect of a socio-technical evaluation.

All in all, our analysis points to the following dilemma: While separation and modularization of tasks is a successful principle with respect to engineering projects, it is not applicable to the design of socio-technical concepts. We intend to conduct more research on this dilemma, aiming at identifying critical success factors for socio-technical concepts. Existing literature on continuous improvement processes (e.g. [26]) could be a potential starting point for this research.

6 References

1. Baxter, G., Sommerville, I.: Socio-technical systems: From design methods to systems engineering. Interacting with Computers. 23, 1, 4–17 (2011).
2. Carroll, J.M.: Scenario-Based Design for Human Computer Interaction. John Wiley & Sons Inc, New York et. al. (1995).
3. Cherns, A.: Principles of Sociotechnical Design Revisted. Human Relations. 40, 3, 153–162 (1987).
4. David.J, K.: Einführung in die allgemeine Systemtheorie. W Fink, München (1996).
5. Diehl, M., Stroebe, W.: Productivity loss in brainstorming groups: toward the solution of a riddle. Journal of personality and social psychology. 53, 3, 497–509 (1987).
6. Dumas, M. et al.: Fundamentals of Business Process Management. Springer (2013).
7. Eason, K.: Information Technology and Organisational Change. Taylor and Francis, London (1988).
8. Eric, S.: Social modeling for requirements engineering. Mit Press (2011).
9. Fischer, G., Herrmann, T.: Socio-Technical Systems: A Meta-Design Perspective. International Journal for Sociotechnology and Knowledge Development. 3, 1, 1–33 (2011).
10. Fischer, G., Ostwald, J.: Seeding, evolutionary growth, and reseeding: Enriching participatory design with informed participation. Proceedings of the Participatory Design Conference, PDC. pp. 135–143 (2002).
11. Floyd, C. et al.: STEPS to software development with users. ESEC'89. pp. 48–64 Springer (1989).
12. Herrmann, T.: SeeMe in a nutshell - the semi-structured, socio-technical Modeling Method. (2006).
13. Herrmann, T. et al.: Socio-technical Self-Description as a Means for Projects of Introducing Computer Supported Cooperation. HICSS. pp. 232–241 (2007).
14. Herrmann, T.: Systems Design with the Socio-Technical Walkthrough. In: Whitworth, B. and de Moor, A. (eds.) Handbook of Research on Socio-Technical Design and Social Networking Systems. Information Science Reference (2009).
15. Huff, J.O. et al.: Strategic renewal and the interaction of cumulative stress and inertia. Strategic Management Journal. 13, S1, 55–75 (1992).
16. Jones, S., Maiden, N.A.M.: RESCUE: An integrated method for specifying requirements for complex socio-technical systems. In: Mate, J.S. and Silva, A. (eds.) Requirements Engineering for Sociotechnical Systems. pp. 245–265 Idea Group, London (2004).
17. Loser, K.-U. et al.: A Drifting Service Development: Applying Sociotechnical Design in an Ambient Assisted Living Project. In: Viscusi, G. et al. (eds.) Phenomenology, Organi-

zational Politics, and IT Design: The Social Study of Information Systems. pp. 311–323 IGI Global (2012).
18. Luhmann, N.: Social Systems. Stanford University Press, California (1995).
19. Mark, G.: Conventions and commitments in distributed CSCW groups. Computer Supported Cooperative Work (CSCW). 11, 3-4, 349–387 (2002).
20. Mate, J.L., Silva, A.: Requirements engineering for sociotechnical systems. IGI Global (2005).
21. Mumford, E.: Effective systems design and requirements analysis: the ETHICS approach. Macmillan (1995).
22. Nierhoff, J. et al.: KreativBarometer: Disclosing the dynamics of creativity climates. „interaktiv informiert – allgegenwärtig & allumfassend!?“ - Tagungsband. (2012).
23. Nolte, A., Prilla, M.: Anyone can use models: Potentials, requirements and support for non-expert model interaction. International Journal of e-Collaboration. Special issue on “Collaborative usage and development of models.” 9, 4, (2013).
24. Prilla, M. et al.: Collaborative Reflection at Work: Supporting Informal Learning at a Healthcare Workplace. Proceedings of the ACM International Conference on Supporting Group (GROUP 2012). (2012).
25. Rittgen, P.: Collaborative Modeling: Roles, Activities and Team Organization. International Journal of Information System Modeling and Design (IJISMD). 1, 3, 1–19 (2010).
26. Spear, S.J.: The Toyota production system: an example of managing complex social/technical systems. 5 Rules for designing, operating, and improving activities, activity-connections, and flow-paths. Harvard University (1999).