‘Reflective User’ in Practice: Explorations from two cases

Samuli Pekkola
Department of Computer Science and Information Systems
PO Box 35 (Agora), 40014 University of Jyväskylä, Finland
samuli@cc.jyu.fi

1 Introduction

Information systems development (ISD) methodologies are numerous (Iivari et al. 2001). Yet, they do not address the change in the information system itself when it is introduced into an organization, or when the organization or its environment changes (Lyytinen 1986; de Michelis et al. 1998). It can even be said that the information systems development begins when it is introduced into an organization (Nurminen and Forsman 1994).

One reason for the deficiencies in ISD methods and obscurities in systems development is the difficulty of anticipating its use in the working environment (c.f. (Robinson 1993). As a consequence, it is very difficult for systems developers to create complete use cases or make appropriate design decisions. Instead, they have to rely on end-users and consider them as the sources of information and most important factors in successful systems development (c.f. Lynch and Gregor 2004). In other words, an input from the user is used to validate the appropriateness of the design decisions.

But this is not an easy task. Evaluating the appropriateness is highly subjective – the same design can be perceived correct for one, and incorrect for the other. The situation is even more difficult if the users do not know when, how, in which context, and with whom they would use the system – which is often the case with research prototypes. In this position paper, I present explorations from two different cases where the users’ suggestions for the features of CSCW systems, and the

feedback for the appropriateness of the design decisions are found problematic to identify and articulate. The first case is composed of two research projects where CSCW applications were developed to support communication and cooperation within an organization. The second case is about a research project where tools for inter-organizational cooperation and communication were investigated.

2 Case 1: Support for intra-organizational activities

The support for intra-organizational communication and collaboration was studied in two consecutive research projects: VIVA (1998-2001) and MADE (2002-2003). In both projects, the aim was to develop a system to support the work of mechanical engineers of a manufacturing company. The engineers worked with others that were located in another site of the company, in their customer's site, or were traveling between them. Meanwhile, because of the context and the risks for enormous financial losses, it was essential that appropriate engineers can be contacted and information shared regardless of their location. The engineers had worked this way for years sharing information through email and telephone and, importantly to the developers, recognizing the problems in real-time communication. Consequently they had a lot of expectations from the future system, which they expected to be able to solve many problems of email and telephone. The need was very practical and very concrete – to share information and communicate with others in real-time. The projects were established for answer this need; first VIVA for PC's, and then MADE for mobile terminals.

For the applications, there was no particular purpose or context for which they were targeted (apart from information sharing and communication). The idea behind there was to offer multiple communication and collaboration tools, e.g. text chat, audio, shared whiteboard, shared text editor, file transfer and short messages so that the users could choose the device, or a set of many that they found the most appropriate at any particular moment (c.f. Pekkola 2003; Pekkola et al. 2003). Both VIVA and MADE systems succeeded in term of achieving the goal.

The systems development lifecycles followed evolutionary prototyping approach (McConnell 1996). Seven different prototype versions were designed, implemented and evaluated in the company. Consequently, the users' reflections on the appropriateness of the design were concerned. These occurred through joint design workshops, researchers observing the work situations, making interviews, performing paper prototype evaluations and log-file analysis, and getting direct feedback to the developers. Different methods had dissimilar benefits: prototypes concretized the design ideas; workshops provided a method to establish a 'common language' and to commonly understand the work processes and general requirements; work situation observations revealed some unspoken issues of work;

interviews gave detailed requirements and design suggestions; paper prototypes validated UI designs; and log-file analysis of usage and direct feedback grounded the comments. During the development process, when the users began to see the benefits of the system for their work, they started to propose further improvements, which could make the system even better. In fact, their number increased with the quality (i.e. lesser errors) – just as Prinz et al. (1998) proposed. The value of the systems can be expressed in the following quote from one of the users of VIVA:

> "We could work with text chat, since it has been used and tested so much that we know it thoroughly, and know how to apply it to different situations. And with [tele]phone, which does the same thing [as text chat] but only quicker. Audio [in VIVA], however, is good if there are three participants. […]. But after all, I think the added-value with VIVA is the combination of different media." (MT, autumn 2001)

3 Case 2: Support for inter-organizational activities

Another project, TechMedia (2003), focuses on inter-organizational communication and information sharing. The project aims at developing ICT solutions to support networked business operations between a manufacturer and their customer on a factory floor level. In other words, the project tries to support cooperation between groups of experts in two organizations with different objectives, strategies, cultures, operations, practices, and technologies, among other things.

The manufacturer is the same as in the Case 1. However, now the activities are not as time critical as earlier, since the manufacturer is offering only support services to the customers. These include, for instance, maintenance and repairing services, analysis of problems or potential problems, and fine-tuning and modifications for improved performance of the machine they produce. These services are exploited in routine maintenance and minor problem-fixing operations as well as when planning larger maintenance maneuvers.

But this is still to come. Currently, the manufacturer has no means to monitor how their suggestions and services are considered. They can monitor a part of the customer's information systems, but not the whole, making it difficult for the manufacturer to understand the context and identify causal-relationships. These make it difficult to distinguish the benefits of the services so that they can be improved, and more importantly, so that they can be sold to others. On the other hand, the customer does not know how to make the most of the service – their processes and information systems do not meet this objective. They have done their industrial business without such services for dozens of years. Nevertheless, both the manufacturer and their customer agree that there is a desperate need for these services to keep them on the market. (c.f. Heikkilä et al. 2005).

In TechMedia project, the objective is to support and encourage knowledge management between the manufacturer's and customer's employees at the factory floor level. This is approached by implementing a shared workspace where the reports on monitoring and analysis are uploaded, their statuses are monitored, and related discussions held.

The development process was far from straightforward. At the beginning, none of the parties; manufacturer or their customer, had any concrete idea what the outcome would be. This was because no one knew the (business) process to be supported, the future users, the infrastructure where the system would operate, or what information was needed to be shared. Even the need for such a system was initially questioned by the customer. In other words, all traditional points of reference were missing. However, after organizing numerous interviews and workshops with different parties, it became possible to create a mutual understanding of the problems, challenges, and solutions of a new business model.

In contrary to the Case 1, here the user is less reflective but more participative. The appropriateness of the design solutions are still to be seen as the work is in progress, and for instance, the (business) process is not fixed yet. Still, because the users participated in the design and even acted as system designers, it is expected that the results are to be validated and approved, as Lynch and Gregor discussed (2004). It is expected that overall design is appropriate, although some details might get changed in the future.

4 Discussions

Here I have presented explorations from two different cases where the users were involved as 'reflectors' for the appropriateness of design decisions. Can a 'reflective' user be defined accordingly?

In the first case, the research was closer to systems development. This means that unknown variables are few. The users know what they want (on a large scale), they are known, there is a certain process to be supported no matter how vaguely it is defined, information flows are more or less defined, and underlying infrastructure is known. The second case is further away from development as the situation is more complex – basically nothing is known. There the user cannot evaluate the design precisely. So, for the users to be 'reflective', the development project must be closer to 'development' rather than 'research'. Users, conditions, processes and environment have to be simply known.

Both cases (and all the projects) followed action research approach where the researchers attempt to alter the object of the study. In Case 1, the researchers were more distant to the users while in Case 2 they cooperated very tightly with the users. To compare these modes of cooperation from the reflective user point of view, in Case 1 the users were equally active but more reflective while in Case 2 they were more participative. In other words, for users to be reflective, they cannot be too engaged with the developers. This minimizes their reciprocal

influence and provides a ground for objective evaluations of the appropriateness of the design decisions. If the users are 'too' engaged, their role will change to as participants influencing, and not validating, the design.

In Case 2, the number of unknown factors was extensive. Business process, the users, technological infrastructure and information-flows were all unknown and undefined (in fact some of them are still so after two-thirds of the project). This led to tight collaboration with potential users. However, regardless of this kind of intensive cooperation, one can question whether the users can really be relied on. Each user, being reflective or not, looks at the situation from his/her perspective, with his/her own experiences, knowledge, education, history, and tasks. This personal background is influenced by the organizational issues such as organization's objectives, strategies, cultures, operations, practices, and technologies, among others. Hence, if the user can provide comments about the appropriateness of the design, they are more likely to be biased especially in unfamiliar cases and situations. In Case 1, the users were able to do so as they were familiar with the objectives. In Case 2, this is less likely to happen as the number of variables is much greater.

5 References

de Michelis, G., E. Dubois, M. Jarke, F. Matthes, J. Mylopoulos, J. W. Schmidt, C. Woo & E. Yu (1998). 'A Three-faceted View of Information Systems.' *CACM* 41(12): 64-70.

Heikkilä, J., M. Heikkilä, J. Lehmonen & S. Pekkola (2005). 'Smart ICT support for business networks'. Vervest, van Heck, Preiss & Pau (eds) *Smart business networks*. Springer: 389-403.

Iivari, J., R. Hirscheim & H. K. Klein (2001). 'A Dynamic Framework for Classifying Information Systems Development Methodologies and Approaches.' *JMIS* 17(3): 179-218.

Lynch, T. & S. Gregor (2004). 'User Participation in Decision Support Systems Development: Influencing system outcomes.' *European Journal of Information Systems* 13: 286-301.

Lyytinen, K. (1986). *Information Systems Development as Social Action: Framework and Critica*l *Implications*. Jyväskylä, University of Jyväskylä. McConnell, S. (1996). *Rapid Development*, Microsoft Press. Nurminen, M. I. & U. Forsman (1994). 'Reversed Quality Life Cycle Model'. Bradley & Hendrick (eds.) *Human Factors in Organizational Design and Management IV*. Elsevier: 393-398.

Pekkola, S. (2003). *Multiple media in group work: emphasising individual users in distributed and real-time CSCW systems*. Jyväskylä Studies in Computing. Jyväskylä.

Pekkola, S., A. Rikalainen, T. Toivonen, S. Hujala, N. Kaarilahti, R. Lintinen & P. Pohjola (2003). 'MADE - A Groupware Application to Support Real-time Activities of Distributed and Co-operating Communities'. *Groupware: Design, Implementation, and Use -9th International Workshop (CRIWG 2003)*, Springer Lecture Notes in Computer Science 2806.

Prinz, W., G. Mark & U. Pankoke-Babatz (1998). 'Designing Groupware for Congruency in Use'. In: *CSCW'98*, ACM Press. Robinson, M. (1993). 'Design for unanticipated use ....". In: *ECSCW'93*, Kluwer Academic Publishers.