Sustaining Participatory Design in the Organization - Infrastructuring with Participatory Design

Johan Bolmsten

Software and System Section

IT University of Copenhagen

A thesis submitted for the degree of

Doctor of Philosophy

September 2015

Tillägnad min kära familj

Acknowledgements

I would like to express my sincere and deep gratitude to my supervisor Dr. Yvonne Dittrich. Yvonne has described how it interests her to take on research in unchartered territories – “in the wild”. I think this PhD study about sustaining Participatory Design in an organization based on an embedded action research approach meets this criterion. This has made the PhD study rewarding in terms of new insights, but also challenging. It would not have been possible to carry out without Yvonne's mentoring and expertise. She has a capability to listen, critically reflect with people based on their perspectives, and creatively engage with them in taking further steps. This requires both skills and courage, and is something that I hope I have been able to learn.

My gratitude also goes to the members of my doctoral committee: Dr. Dag Svanæs, Dr. Teija Helena Karasti, and Dr. Keld Bødker for their insightful comments and suggestions.

This thesis would not have been possible without the staff at the World Maritime University who have supported me and contributed with their perspectives and engagement throughout the research. One colleague recently said: WMU combines two great things, education and a truly intercultural setting, where students and staff come from all corners of the world, bringing with them not only different national cultures, but also different professional and social traits. This diversity has been important to my research, and I am grateful to WMU for letting me be part of this environment.

Abstract

Modern organizations need to be able to change to seize opportunities and meet challenges, which are ever more rapidly presenting themselves. In doing so, they need to make use of the creativity and innovations of their employees. At the same time Information Technology applications today are likely to take the form of complex, integrated infrastructures, supporting collaboration within and across organizations. This places requirements on the development of IT infrastructures. As the work practices within an organization change, the supporting infrastructure also needs to evolve. This PhD thesis is about sustaining Participatory Design in the organization to enable users to influence the development of the IT infrastructure that supports their work practices.

The empirical research is based on a long-term action research study, where this researcher works as an embedded researcher, complementing action research with daily work of developing software support with users. In addition, ethnographically inspired research methods have been used to understand and evaluate how different situated development practices come together in infrastructure development.

The empirical results put forward the following contributions: (1) Shop floor IT management is a core capability for innovation, and is a driver for sustained PD in the organization. Users on the shop floor trigger infrastructure development when their IT applications need to be technically and organizationally integrated. (2) A flexible technical infrastructure is needed to support Participatory Design. The technical base as infrastructure both enables and constrains the design of local software support, as well as the application of Participatory Design methods. (3) Users on the shop floor need to participate in organizational IT management in order to relate the development of their local software support in an integrated infrastructure. The results of the action research report four interlinked improvements to sustain Participatory Design in the organization concerning structuring end-user influence in the organizational arena, a participatory and evolutionary project management, and participatory tools and techniques appropriated for infrastructure development.

Table of Contents

1. Introduction

Modern organizations need to be able to change to seize opportunities and meet challenges, which are ever more rapidly presenting themselves. In doing so, they need to make use of the creativity and innovations of their employees. At the same time Information Technology (IT) applications today are likely to take the form of complex, integrated infrastructures, supporting collaboration within and across organizations. This places requirements on the development of IT infrastructures. As the work practices within an organisation change, the supporting infrastructure also needs to evolve. This PhD thesis is about the usefulness of Participatory Design (PD) to enable users in the organization to influence the development of the IT infrastructure that supports their work practices. PD is an apex approach to user-centred design, which is already making valuable contributions to today's organizations in managing design projects and putting forward tools and techniques to negotiate design between users and IT-professionals (see, for example, Bødker, Kensing, and Simonsen, 2004, MUST method and knowledge work, which is also used in this research). To take on infrastructure development, this PhD research expands the application of PD beyond local projects, moving towards a sustainable participatory IT management in the organization.

Sustained PD is an emerging research theme in the PD research community that is positioned by Simonsen and Hertzum (2012) and that prompts PD to "think big" and take on both the design and implementation of larger Information Systems in organizations. This entails a number of new challenges for PD: how to create appropriate organizational conditions for a PD approach of development; how to manage a multitude of stakeholders; and how to manage a stepwise implementation process. This PhD research targets these challenges. In addition, to develop a PD approach that incorporates design as well as implementation and that goes beyond the local project, this thesis relates to PD research about shop floor IT management (Eriksén, 1998), which also includes End-User Development (see, for example, Dittrich, Eriksén, & Hansson, 2002).

To both understand the challenges and develop improvements of how to sustain PD in the organization, this thesis also makes use of concurrent research contributions of "infrastructuring" coined by Karasti and Syrjänen (2004) and Karasti and Baker (2004). Infrastructuring is about developing a sensitivity to understanding communities taking ongoing responsibility for the participatory development of software and work practices over the long-term. The use of infrastructure provides an analytical focus that goes beyond the production of an individual piece of software, and foregrounds an understanding of the socio-technical relations where it is embedded. In this thesis, infrastructuring provides an analytical framework to understanding how situated shop floor IT management practices connect to supporting technical and organizational infrastructure on which it necessarily depends.

The overall research question that has guided the study is:

How can end-users participate in the evolution of an organization's IT-infrastructures?

This research question has its origin in practice and underpins an action research and ethnographic PhD research study. The setting for the empirical research is the World Maritime University (WMU), UN, in Malmö, Sweden. Users working at WMU are accustomed to taking charge of the development of IT to support their work – with or without the assistance of IT-professionals. Three such cases are reported in this study, ranging from: 1) faculty and faculty assistants working closely with IT-professionals in the development of course administration support (such as scheduling, marking, and e-learning components), 2) to registry staff also taking on the technical development of a registry system to support enrollment, grade reporting, curriculum quality evaluation, and student welfare and living support, and 3) an administrative assistant that is developing electronic forms and an address database. These shop floor IT management practices, where local software development takes place in close connection to daily work activities in different situated constituencies, are established approaches that predate the research study by many years.

At the same time, there is an increasing need for cross-organizational collaboration and integration – this is what forms the basis of practical contributions of the action research. The enrollment process, for example, not only takes place within the registry department but has many points of integration with the faculty, where information flows back and forth, and many considerations and decisions have to be made on both ends before a student is enrolled. In the same way, marking entails a work process that first involves a number of faculty and faculty assistants, and later continues at the registry department. Additionally, the working purpose of the electronic forms is not only for them to be used by the administrative department alone, but by all departments and published in common information repositories.

The empirical research is based on a long-term action research study, where this researcher works as an embedded researcher, employed as an IT-Professional by WMU and complementing action research with daily work of developing software support with users. The research is methodologically structured by Dittrich, Rönkkö, Eriksson, Hansson, and Lindberg's (2008) Cooperative Method Development (CMD) approach. CMD is based on an action research cycle, where users participate in both the development of improvements and evaluation of the results. In addition, ethnographically inspired research methods have been used to understand and evaluate how different situated shop floor development constituencies can come together in development activities.

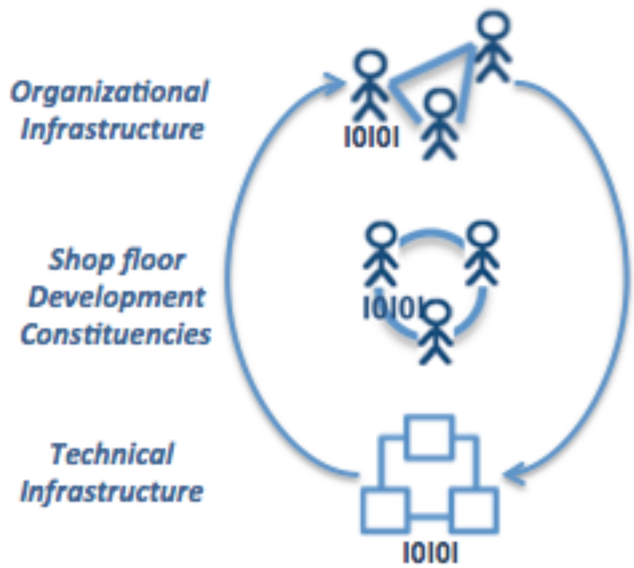


Figure 1.1 Shop floor development constituencies supported by technical and organizational infrastructure in sustaining PD in the organization.

In answering the research question, the empirical results put forward the following contributions, which are empirically described and evaluated in chapters 5-9 and discussed together in chapter 10:

- The first contribution is an explication of how *Shop floor IT management can act as the driver for sustained PD* in the organization. The importance of shop floor IT management is shown in order to take advantage of the innovatory design capabilities of users working in an organization and for them to develop useful IT applications. In working with a knowledge intensive and culturally heterogonous organization, this was a key dimension to understanding the development of software support, which in a number of cases can be seen as related to core capabilities in the organization. A red line through the empirical results is that and how users engage in shop floor IT management, both as participating users and as end-user developers. The empirical research also indicates how users take on an expanding role as technical developers, with and without the support of IT-professionals. Users on the shop floor trigger and participate in infrastructure development when their IT applications need to be technically and organizationally integrated.

- A flexible *technical infrastructure* is needed both for IT-professionals to support user-oriented software development, as well as for supporting shop floor IT management and End-User Development. The second contribution from the empirical material shows how technical bases as infrastructure matter in enabling and constraining not only the design of local software support but also the application of PD methods. In the action research of this thesis, the need for an integrated technical base was the raison d'etre for applying PD beyond the local project. Through a number of empirical cases, it is shown that if PD is going to be organizationally viable according to core principles of user influence, the technical base cannot be black-boxed and needs to be included in the realm of PD concerns. To relate the development of different IT applications, shop floor users need to participate in organizational participatory IT management. The empirical results show how shop floor users can manage the technical and organizational infrastructure that frames their local shop development constituencies.
- The improvements put forward by the action research relate to structures, procedures, and representations to organizationally manage IT infrastructure development. The first improvement relates to the importance of structuring end-user influence on the organizational arena, where a committee-based management is developed to enable users to influence organizational IT management. The second improvement regards a participatory and evolutionary project management that also connects to strategic organizational management. Effective participation in organizational IT management becomes important to manage developments that span across several local IT

applications. Finally, the third improvement concerns the appropriation of PD tools and techniques to support users to negotiate matters pertaining to infrastructure development.

With this chapter serving as the introduction, the rest of this thesis is structurally organized as follows. The next two chapters (2 and 3) review related work. The research approach and an overview of the empirical research are described in chapter 4. The following five chapters report and evaluate the empirical research in detail (chapter 5, 6, 7, 8 and 9). The outcome of the empirical research is then jointly reported in chapter 10, followed by a concluding chapter.

2 Participatory Design in the organization

A main contribution from Participatory Design (PD) to the development of Information systems (IS) comes from partial translations (Suchman, 2002) between the different professional worlds and associated social and technological aspects of users and IT-professionals. PD is a long-term established design approach of user-oriented software development with a focus on "located accountabilities" (Suchman 2002) of design of useful system support, where “design[ed] from somewhere” is targeted as opposed to the production of discrete devices “design[ed] from nowhere.” In this way, PD is part of a growing process-oriented software development community that takes a stance in situated action and a locally accountable design of software support, which also forms an alternative towards prevalent IS approaches. In this thesis, PD is positioned as a comprehensive approach to manage design and implementation of IS in the organization. The use of PD moves beyond design in the local project towards a sustained organizational approach. This is an emergent research track in the PD community.

This chapter starts with the account of relevant core principles and assumptions of PD research. In addition, different generations of PD research are described, and how this thesis responds to the call for a new generation of PD research that (again) focuses on the organizational arena. The second section introduces the use of, and need to amend, PD tools and techniques in local project development. The third and fourth sections positions shop floor IT management and End-User Development (EUD) as the foundation to understand PD that extends beyond a local design project context. The fifth section portrays how an evolutionary development approach to PD targets both design and implementation. Finally, the sixth section points to an emergent research track to sustain PD in the organization, to that this PhD research contributes.

2.1 Participatory Design foundations and history

There are core common principles and design assumptions that underpin PD and that have evolved together in different generations of PD research. When initiating the research and being on the outlook for an appropriate guidance in related work to support the action research, principles of PD came across as coinciding with the in-situ rationale of how World Maritime University (WMU) went about the development of software support:

1. *Workers are intelligent, creative, and productive contributors to organizations if they are empowered to express their expertise, exercise their decision-making capabilities, and given responsibility for the impact of their decision-making.*
2. *PD holds that, contrary to a Tayloristic belief, good ideas are as likely (perhaps even more so) to come from the bottom up as from the top down. Front-line workers know what works, and have a lot of ideas of how to improve things* (Miller, 1993).

From the point of view of this thesis, these principles can be related to PD assumptions of knowledge and power diversity in design. From an epistemological point of view, PD asserts a pluralistic view of organizational actors and their interactions as inherent characteristics. Knowledge is seen as closely related to actions, and can be regarded as a function of a particular stance or perspective. This implicates that the design of useful system support comes to carry a located accountability related to the situated actions of a person carrying out a given set of work activities (Suchman, 1994a). By default, a user can thereby be recognized as a domain expert of her / his particular work activities, one who know what works and what will serve as a primary source of ideas of how to improve things.

This is the point of departure in the application of PD in this thesis. At the same time, when applying the above reasoning, an unavoidable consequence for design is that different stakeholders have different "perspectives" of the application domain, which entail blindness and bounded rationality. This is overcome by self-reference through active cooperation between different stakeholders. This is also a process of negotiation between different interests. Different user groups need to come together with IT-professionals and other stakeholders to negotiate a new design of software support, where individual perspectives and stances can both complement, as well as potentially be in conflict with, each other. To this end, different stakeholders not only bring different sets of knowledge to the table but also different interests and positions of power. In recognizing this, PD from the outset has incorporated a dimension of recognizing and coping with conflict in design activities.

This contrasts PD to user-oriented development approaches that are based on a consensus stance between different stakeholders, such as, for example, the socio-technical design tradition (Bjerknes, Ehn, Kyng, & Nygaard, 1987). In the socio-technical design tradition, power considerations are argued flawed, as workers are on the same side as managers in the pursuit of one common good in the organization. In coping with power, PD tools and techniques on the other hand traditionally strive towards creating an egalitarian mode of collaboration in order to promote the voice of users to be heard in relation to managers and IT-professionals. In other words, a level playing field, where facts, experiences, and meaningful concepts viewed from different perspectives of the application domain are represented. Since early PD research, notions such as Bråten's (1973) model monopoly are used to understand how the perspectives of IT-professionals and managers dominate the design process on the expense of the perspectives of users.

In the empirical domain of this thesis, power diversity is still very much a relevant dimension, but the traditional top-down vs bottom-up dichotomy in PD is diversified. As is further described below, knowledge and power together become connected as a dynamic development component of the organization.

The focus of PD research in relation to knowledge and power diversity assumptions in design has varied. There are a number of inside, outside, and mixed accounts of the history and applications of PD (see for example Bjerknes et al., 1987; Blomberg & Kensing, 1998; Clement & van den Besselaar, 1993; Floyd, Mehl, Reisin, Schmidt, & Wolf, 1989; Schuler & Namioka, 1993). To this end, different generations of PD research are recognized by, among others, Blomberg and Kensing (1998), Gärtner and Wagner (1996), and Floyd, Reisin, Schmidt, and Wolf (1989) These generations do not define all PD research contributions during a certain period, but cluster broad trends of research.

First generation PD research included both a knowledge and power dimension that incorporated the local project, the organization, and even the national arena (Blomberg & Kensing, 1998; Gärtner & Wagner, 1996). Early and influential PD projects, such as NJMF or DEMOS, conducted during the 1970s were often anchored in a search for humanization and ethics at the workplace and were often conducted in collaboration with unions. These projects not only took a stance against control-oriented design approaches that were recognized to apply top-down "tayloristic" industrial engineering principles to rationalize and synthesize work activities in the organization (that can also be related to the IS management approaches of today). These projects also took a stance against the consensus or harmony oriented nature of Socio-Technical Systems approaches by placing greater emphasis on the idea of democratization and the negotiation between conflicts of interests in power structures and hierarchies (Floyd et al., 1989).

Second generation PD research has continued to evolve these principles in regard to tools and techniques to support users and IT-professionals in a knowledge building process in the design of software support, predominantly in a local project setting. (This PD research track is further illustrated in the context of this thesis in the following section.) However, the need for a new generation of PD that again targets the organizational arena has been well noted for some time in the PD community. Clement and Besselaar (1993), for example, note that PD can be characterized by isolated projects with few signs of becoming self-sustaining work processes within the implemented work settings. An explanation of this state is offered by Kensing and Blomberg (1998), who put focus on the issue that PD projects have during the last 20 years "somewhat focused on the individual project arena where specific systems are designed." According to Gärtner and Wagner (1996), emphasis has been put on how to foster a "direct and unmediated partnership between designers and the users of systems." They consider that the main challenge for PD is to deal with the broader organizational arena "on which Participatory Design initiatives depend for their long-term survival."

New openings for PD on the organizational arena can be related to a recognition that the traditional division and conflict between managers and workers are being diversified. Clement and Besselaar (1993) and Bødker (1996), for example, note that where historically the problem to be solved was that of "rude exploitation" from management, the situation today is more vague and does not lend itself to such easily identified conflict. At the same time, new conditions for user participation in the organization are emerging "where the cons are set for the users to participate with designers (and managers)" and a re-interpreted usage of the "Scandinavian collective resource projects can help research in this process" (Bødker, 1996).

New alliances between groups in organizations are suggested "with due concern for their diversity of resources, and with constructive use of the conflicts inherent in the organization – can be a way forward in empowering organizations, making room for groups and individuals within them to act" (Bødker, 1996). The action research in this PhD research contributes to PD research efforts that are reclaiming the organizational arena with an updated take on knowledge and power diversity. Concurrent PD research tracks to this end are further described in section 2.6.

2.2 Participatory Design tools and techniques for users and professional designers

Second generation PD research (Floyd et al., 1989, p. 285) focuses on how to support situated design work between users and IT-professionals in local design projects. If the main idea of the first generation PD projects was to support "democratic planning" in the organizational and even national arena, the latter generation has come to focus on the idea of designing "tools for skilled workers" (Ehn & Kyng, 1987) to enable participation in situated design processes in the individual project arena (Blomberg & Kensing, 1998; Gärtner & Wagner, 1996). This has also entailed (and allowed for) a re-orientation of PD research from a union context to a business context. How a PD strategy can be practically applicable in a business-oriented organizational setting is exemplified by Bødker, Kensing, and Simonsen's (2004) MUST method.

PD in the local project to support design between users and IT-professionals has been of importance in this PhD research. As a concrete reference, the MUST approach, together with other sources of documentation of the application of PD tools and techniques found in Greenbaum and Kyng (1992) and Kyng (1995), have been used as a base to guide the action research in the local projects at WMU.

The MUST method features both a project management approach and knowledge framework that positions the usage of PD tools and techniques. These components are denoted by four principles (that can be related to the core PD assumptions described above): (1) the principle of a coherent vision for change targets that a sustainable design should be considered from the user's perspective that includes IT-development, organizational development, and qualifications development; (2) the principle of genuine user participation puts a focus on the importance of actively involving users in the design process of their system support; (3) the principle of firsthand work experience points out that IT-professionals should not base their system design work on abstract descriptions alone, but benefit from engaging in the work domain of the users directly to design a useful system; and (4) the principle of anchoring visions denotes that for a successful project outcome, all stakeholders, including both users and management, need to acknowledge a design project's goals, visions, and plans.

To put these design principles in practice, the MUST project management approach consists of four phases: (1) the initiation phase where a project is established; (2) an in-line analysis that aligns the design project's goals and the company's business and IT-strategies; (3) an in-depth analysis that focuses on understanding selected work practices; and (4) an innovation phase that outlines visions of future IT-systems together with associated work organization improvements.

Furthermore, the knowledge framework (Table 2.1) in the MUST method indicates where a local design project needs to generate knowledge and positions the use of PD tools and techniques. It gives an understanding of where users need to learn about available technical options and where IT-professionals need to acquire knowledge about the work practices in the application domain. In accordance with how a “useful” design of systems is anchored in the situated actions of users, this knowledge is gained in the interplay between concrete experience (direct firsthand experience of a given technical or organizational area) and abstract knowledge (expressed via descriptions and models of the respective area of knowledge). In greater detail, concrete knowledge is gained through, for example, users being presented with exemplary systems and IT-professionals carrying out workplace visits. Abstract knowledge is made up of, for example, functional models with relevant descriptions of users’ work practices. Common boundary objects, such as mock-ups and prototypes, can then be used to mutually explore visions and design proposals of new IT design usage.

	Users' present work practices	**New IT usage**	**Technological options**
Abstract knowledge	Relevant descriptions of users' present work practices	Visions and design proposals	Overview of technological options
Concrete experience	Concrete experience with users' present work practices	Concrete experience with the new IT usage	Concrete experience with the technological options

Table 2.1. Bødker et al.'s (2004) knowledge framework to position PD tools and techniques where a design project needs to develop knowledge.

The use and development of the MUST method in this thesis will be described in further detail in especially chapters 8 and 9. The need to further develop MUST also comes to illustrate the research challenge of this thesis. Although the MUST approach begins to address the relation between the local project and the organizational arena through the inline analysis phase, it is predominantly focused on the former. In addition, MUST is focused on design, but to sustain PD in the organization, implementation and maintenance need to be considered as well. In the same way that MUST as a PD approach combines a focus on both situated action and planning in the local design project context, there is the need to go beyond the local design project to sustain PD in the organization. Related research on PD beyond the local design project, as well as emerging research on sustaining PD in the organization, to which this thesis contributes, is further described in the forthcoming sections.

2.3 Shop floor IT management

The established practices at WMU in development of software support can be related to in terms of a shop floor IT management approach (Eriksén, 1998). Shop floor IT management can be recognized as the foundation of how this research contributes to a PD approach in the organizational arena. End-users, with or without the support of IT-professionals, have traditionally managed the development of their own software support. In the context of this thesis, their shop floor IT management can be connected to Dittrich, Eriksén, and Hansson's (2002) "PD in the wild" perspective, where "design in use" is positioned to complement a common "use in design" (Bødker, 1999) research stance of PD research.

This diversifies a traditional distinction between use and design that has also been present in PD. Traditionally, PD tools and techniques have focused on bringing "use into design" (Bødker, 1999) as a reaction to predominate software engineering approaches. This is the case in, for example, the MUST approach, as accounted for in the previous section. In the two cases studied by Dittrich et al. (2002), it was found that important design practices of interpretation, appropriation, assembly, tailoring, and further development of computer support, were carried out in what is normally regarded as deployment or use. This both involved users working together with IT-professionals, as well as users taking responsibility for the design of software as end-user developers (further described in the following section). Development in this way takes place in interlaced design constituencies, (Wessels, Walsh, and Adam, 2008) - in this PhD research referred to as shop floor development constituencies - i.e. assemblies of different stakeholders who are entitled through their interest, role, or expertise to contribute to specific design and development activities. Both the design constituencies and the organizational affiliation of specific participants are subject to situated negotiations and decisions.

The notion of shop floor IT management refers to Wenger's (1998) pioneering work concerning Communities of Practice (CoP) (see also Ackerman, Pipek, & Wulf, 2003; Lave & Wenger, 1991; Wenger, McDermott, & Snyder, 2002). CoP are groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis. Over time, they "develop a unique perspective on their topic as well as a body of common knowledge, practices, and approaches. They also develop personal relationships and established ways of interacting" (Wenger et al., 2002). Of relevance to the shop floor IT management perspective developed in this thesis, Fisher (2001) diversifies CoP to include Communities of Interests. An added dimension highlighted in Communities of Interests (Fischer, 2001) is sharing of heterogeneous expertise in a community, not just in one practice. To this end, a shop floor development constituency is made up of a community of people such as users, end-user developers, and IT-professionals that share a strong common domain knowledge and interests, but with different work roles and responsibilities.

In this thesis, it is also of relevance to include a yet unexplored role of Wenger's CoP in relation to Shop floor IT Management. This role is described by Wenger et al.'s (2002) through the notion of the "double-knit" knowledge organization. For an organization to learn from its own experience and to fully leverage its knowledge, the CoP that steward knowledge must be interwoven with processes, procedures, and the intent of structures in creating what can be called a "double-knit" knowledge organization (Wenger et al., 2002, p. 18). Shop floor IT management thereby include both a dimension of situated action and planning where practitioners in dual roles, both as community practitioners and operational team members, help link the capabilities of CoP to the knowledge requirements of teams and business units. In this sense, situated action and planning through processes, procedures, and the intent of structures should work in tandem to promote sustained performance.

Wenger et al. (2002) discuss how the conceptualization and role of management need to change from a traditional management perspective to "cultivation" in order to support the design and development activities of CoPs, where "design and development are more about eliciting and fostering participation than planning, directing, and organizing their activities." If managing to cultivate CoPs, of in this case shop floor IT management, new possibilities are offered for "weaving the organization around knowledge, connecting people, solving problems" (Wenger et al., 2002). In this way, in this PhD research, "management" is a relevant conceptualization, but in a different kind of way. What from the outset distinguishes the empirical domain of this research is that the shop floor development constituencies participate in management of IT also in the organizational arena. In this way, the role of both the research and the action is to support the shop floor development constituencies to cultivate their own management.

2.4 End-User Development

In the PD research track that includes shop floor IT management, there are elements that to an increasing degree include EUD. EUD is about users themselves taking on various degrees of technical development.

Early on, support for sharing and cooperation among user communities in developing useful software support has been examined (see e.g. Kahler, 1995). In the past, EUD in PD was, however, often limited in scope and regarded local tinkering and tailoring. Overtime, EUD has expanded and is now a research domain of its own (Costabile, Dittrich, Fischer, & Piccinno 2011). The field of PD has been both conceptually and pragmatically important in developing this arena (Simonsen & Robertson, 2012, p 127). The expansion of EUD also goes hand-in-hand with developments of technical infrastructures that allow for changes and modifications post implementation. For example, instead of custom development, industry solutions allow for, and are even purposefully designed for, users to continue to develop them to fit their needs, once they experience them in use.

Another strategy to work with usefulness dimensions in industry solutions is to postpone some design decisions until the use phase. It is, therefore, often not necessary for end-user developers to develop technical tools from scratch, but instead they can continue to develop technical infrastructures with different levels of interaction with professional designers. In EUD, there are cases that range from tinkering and tailoring to more substantial appropriation (Lieberman, Paternò, Klann, & Wulf, 2006). Examples of the former range from simple programming of mail filters to the design of complex spreadsheet applications. An example of the latter includes the configuration of a large-scale electronic patient record system to fit a specific care unit within a hospital, indicating an industry solution but built as an open system where the users need to "finish off" the design.

Pipek (2006) provides a categorization of cooperative development scenarios that are useful when connecting EUD to PD: shared usage requires the least coordination, and user groups are a self-help feature in both commercial and private contexts. Cooperative tailoring in a shared context provides better possibilities for sharing customizations, but might result in conflicts if changes to an individual tool hinder the sharing of work results. When users tailor the shared tool, they need to negotiate not only the adaptations but also the usage of the common tool.

The last scenario - of particular relevance to this thesis - regards shared infrastructure, which is also the least researched scenario. Here, tailoring results can affect configurations of other systems. The design space for EUD of an individual application is constrained by the interoperability requirements. Heterogeneous user groups are dependent on each other although they neither share a common work practice nor a common tool.

Although being the least researched, the shared infrastructure scenario in regard to EUD has been discussed as early as 1992, where Gantt and Nardi (1992) describe patterns of cooperation between EUD and users of CAD systems. The authors describe the development of formal and semiformal positions in organizations where local developers not only act as 'gurus' – acquiring and sharing knowledge about how to tweak the system on an individual basis – but also as 'gardeners' in maintaining a set of customizations and tailorings for their group or department and continuously enhancing the common work tools, thus improving the productivity of the whole team. Dittrich and Lindeberg (2003) also discuss such a case; in infrastructures supporting data-intensive businesses, such as telecommunication, the flexibility of a specific application can only be deployed when other applications in the same network and the interoperability platform can be tailored accordingly. The importance of combining EUD and professional development activities when evolving such a common infrastructure and support for it is addressed in Eriksson and Dittrich (2007).

In this thesis, the shop floor people take an extended role as EU-developers of infrastructure that can be related to Bødker's (2000) "platform coordinators." This extends Gantt and Nardi's (1992) conceptualization of users as "gardeners," where the term "gardeners" primarily relates to adaptation of software, the notion of "platform coordinators" highlights the user's role in mediating the development of the technical base.

With technologies becoming more and more accessible to users, an increasing engagement of users on a technical design level also becomes possible. In the proceedings of the 2011 IS-EUD symposium (Costabile et al., 2011), many contributions joined useful dimensions referenced through PD with technical development dimensions in EUD. In regard to the technical base discussion in this study, one linkage of relevance relates to the notion of meta-design in EUD (Fischer, 2011). Meta-design is focused on objectives, techniques, and processes to allow users to act also as technical designers. In doing so, meta-design does not provide fixed solutions but frameworks within which users can contribute to the technical development. From the point of view of this study, the technical base as part of an infrastructure becomes a meta-design framework. If software development is, in fact, not only confined to representing "use in design" in the local project setting but also "design in use," then a logical challenge for PD becomes if and how users can also engage in the design of their own meta-design frameworks, including the technical base.

2.5 Participatory and evolutionary system development

Shop floor IT management together with EUD as part of PD makes ongoing and evolutionary design processes between work practices and new technical options into new software support usage a primary concern. The introduction of a software system to an application area changes the very same application area, as well as the perceptions of the problems for which the system was originally introduced. As a program becomes part of, and is embedded in, the very world it models, design and use cannot be treated separately in succeeding phases - as is the case in the classic waterfall model. In this way, software development can be recognized as not starting from predefined problems; instead, development is viewed as a continuous, cooperative, and evolutionary design process involving both users and IT-professionals (Floyd, 1991). These dynamics of design as a process that involves both the design in itself and the unfolding of a problem, along with its corresponding solution, were already realized by Lehman in 1980 (Lehman, 1980).

As is noted in the International Handbook of Participatory Design (Simonsen & Robertson, 2012), one of the early conceptual contributions to PD methods was the work of Floyd and her group at Hamburg University and their STEPS process model. STEPS – Software Technology for Evolutionary Participatory Systems Development (Floyd, Reisin, & Schmidt, 1989b) – is based on empirical studies of systems development practices in local projects and can be regarded as a methodological frame combining PD and Software Engineering. STEPS can be seen as a methodological support to shop floor IT management in the local project that combines both a participatory design and implementation approach. The STEPS process model is based on the above insight that technical construction of software cannot be separated from how the software is used on the shop floor. IS development is understood as an evolving design process that shapes both the technical artifact and its context of use.

In discussing development of software as a human activity, a starting point in Floyd et al. (1989b) and Floyd (1991) is Naur's (1985) view of programming as theory building. Theory building refers to an insight building process where domain experts, IT-professionals, and other stakeholders are participating. Naur (1985) refers to this insight building process as theory building, which is used in early conceptions of PD to distinguish PD from traditional requirement specification driven software engineering. According to Naur (1985), having a tacit theory of the application domain has primacy over such products as program texts, user documentation, and additional documentation such as specifications. These are considered secondary, as they reveal only part of the application domain theory held by the programmers and the users, which cannot conceivably be expressed (Naur, 1985). Having the theory of the program, one can explain how the solution relates to the affairs of the world that it helps to handle, both in terms of human and technical agencies. One can also explain why each part of the program is what it is and is able to respond constructively to any demand for a modification of the program to support the affairs of the world in a new manner.

2.6 Sustained Participatory Design

PD should think big and engage in continuous large-scale IS development in the organization through a sustained PD approach, applied throughout design and organizational implementation. This is how one of the latest research tracks about sustained PD is conceptualized by Simonsen and Hertzum (2013). In this way sustained PD comes to relate to but also transcend the local development focus of STEPS. Simonsen and Hertzum (2013) argue that if PD is characterized by the aim of establishing mutual learning situations between users and designers, “there is a need for a sustained Participatory Design approach that allows the organization to experiment and learn – not only as part of the initial design but also as part of the organizational implementation and use of a technology.” In what can be related to Floyd's et al. (1989b) evolutionary STEPS method, Simonsen and Hertzum extend PD to an iterative development approach by (1) emphasizing PD experiments that transcend traditional prototyping and evaluating systems during real work; (2) incorporating improvisational change management, including anticipated, emergent, and opportunity-based change; and (3) extending initial design and development into a sustained, stepwise implementation that constitutes an overall technology-driven organizational change.

Simonsen and Hertzum’s (2013) sustained PD approach also highlights an organizational planning complement to the situated focus of shop floor IT management. They recognize how emergent and opportunity-based changes are widely noted in PD projects “but there has been surprisingly little focus on managing learning from such changes over longer periods of time.” A sustained PD approach involves iteratively integrating design and development with organizational implementation and real use in: (1) evaluating progress on planned changes, (2) becoming aware of emergent changes, and (3) turning selected emergent changes into opportunity-based changes. While progress on planned changes is a means to ensure that system possibilities get integrated into actual work practices, turning emergent changes into opportunity-based changes is a means to ensure that work practices are changed in relevant ways. In evaluating the results from an ERP case study, Simonsen and Hertzum's (2013) identify four major challenges for (preferably) PD action research are identified in managing such sustained iterative development process:

1. Creating appropriate conditions for PD: It is recognized that a success factor entails both the customer and vendor needing to be motivated and interested in committing to a PD approach. This is an initial challenge, which in the ERP case laid the ground for the close partnership and collaboration required by the sustained PD experiment.
2. Managing a multitude of stakeholders: large-scale information-systems projects are characterized by involving a number of different actors spanning different organizations and different organizational levels. The second challenge is managing and aligning the motivations and interests of this multitude of stakeholders. In relation to Simonsen and Hertzum's (2013) case, they recognized the challenge to comply with the premises set at the national and political levels and by high-level organizational strategies aligning with the different lower levels, arguing that PD with its direct involvement of end-users is an effective means to manage, mesh, and meet these different interests. Traditionally, the focus in PD is about the relation between professional IT-designers and end-users. However, in expanding PD to the organizational arena the inclusion of a broader range of stakeholders becomes necessary, including management. To this end, Simonsen (1997; 1999; 2007) also presents PD research of the practical application of PD tools and techniques to involve top management in IT projects, suggesting PD tools and techniques to work with strategic alignment and business strategy, such as functional analysis and problem mapping.
3. Managing a stepwise implementation process: A third challenge is to effectively manage sustained large-scale iterative PD experiments, forming an overall stepwise implementation process.

4. A fourth challenge concerns the methodological question of how to conduct realistic large-scale PD experiments to evaluate prototype systems during real work, where it is important methodologically to evaluate not only early and quickly, but also to evaluate progress during a longer period of time.

The discussion chapter of this thesis will explicitly relate to how the results of the action research provides new insights to these dimensions, with special attention given to how it is possible to combine the management of planned changes while maintaining a situated shop floor IT management focus. PD management in the organization extends situated shop floor IT management with PD planning on the organizational arena. Shop floor IT management puts a focus on workers managing local design process. Managing sustained PD in the organization, in turn, connects to local development going on in shop floor development constituencies and involves more organizational stakeholders.

Simonsen and Hertzum's (2013) sustained PD approach is the most concrete recent effort found to answer the call for PD to (again) engage on the organizational arena. Other recent and emerging PD research tracks that can be related include Björgvinsson, Ehn, and Hillgren (2010) through their application of "agonism" (Mouffe, 1999) – albeit their case is not in an organizational context. Through the use of agonism, the authors position a framework of ideas to the PD community of how to relate the empowerment of "a multiplicity of voices in the struggle of hegemony and, at the same time, find constitutions that help transform antagonism into agonism, from conflict between enemies to constructive controversies among adversaries who have opposing matters of concern but also accept other views as legitimate." Following, for example, Hardy and Clegg (1996), "agonism" can also be related to in an organizational context. Contrary to predominate management theories, this entails a pluralist model of organizational governance organization, where actors' exercise of power in different forms is a ubiquitous feature (1992). To this end, Björgvinsson et al.'s (2010) agonistic framework becomes interesting in how it can interprets first generation PD research about democratic management in a new organizational landscape. However, in both these cases, it is notable how they are more a beginning than an end, that is, more a call for research than results.

The overwhelming weight of current management theories

It should be recognized that a participatory IT management in an organization is both novel and challenging. Dahl (1984; 1985) writes about the “overwhelming weight of existing institutions and ideologies” in regard to management theories and their instantiation in organizations, which makes it difficult to design and implement alternatives. Although notions such as “empowerment” are receiving a lot of attention, a fundamental problem with traditional management is the lack of pluralist approaches of power distribution and decision-making in the organization. Agyris (2001) compares the current state of predominate management to the emperor’s new clothes in that “managers love empowerment in theory, but the command-and-control model is what they know and trust best.” According to Minett (1992), conventional models of the organization fail to adequately theorize about the concept of power related to management and decision-making in the organization:

- Neo-classical economists and management-oriented organizational theorists have tended to deny the occurrence of power and politics in the firm except as occasional and dysfunctional intrusions (management theory).
- Marxist economists and sociologists have been preoccupied with class struggle in the wider society, rather than within the confines of the firm itself; even when they have focused on the firm itself it has usually been to analyze various macro-system determined modes of managerial control (industrial sociology).

In this way, the conventional image of the organization provides very little room for that power in managing the organization could be asymmetrically, rather than dichotomously, distributed. Furthermore, the first emergence of pluralist models of power in the organization was primarily focused on arguing against the conception of a single, profit-maximizing goal for all enterprises, and also left out the role of power. The human relation schools, for example, neglected power as a variable, and their research on bounded rationality inadequately considered the question of power. Minett purports (1992) that while newer "pluralist" theories represent progress, they tend to evolve in a pragmatic way and consequently suffer from a lack of systematic conceptualization. They recognize the ubiquity of intra-organizational power in enterprises but fail to give a satisfactory account of its origins and distribution. The pluralist model of power can be related to the agonistic view of democratic governance put forward by Hardy and Clegg (1996). It implies a multidirectional view of power, placing actors in an influence-coercion continuum through which they explicitly and implicitly assess their situation and arrive at strategies of how to "go on." Minett (1992) notes that especially in knowledge intensive organizations, pluralistic governance can be positioned as to the relevance of understanding effective organizations.

2.7 Summary

PD has a long-tradition as a user-oriented development approach. Two fundamental assumptions to support this action research that run through PD research relate to knowledge and power diversity. These assumptions not only distinguish PD from traditional planning-driven software engineering, but also from other user-oriented development approaches, such as the consensus-oriented Socio-technical development approach.

Contemporary PD research has made valuable contributions of tools and techniques to negotiate design between users and IT-professionals in the local design project. A number of such PD tools and techniques are used and appropriated in this PhD research.

This thesis is about PD moving beyond the local design project, and towards a PD approach to comprehensively manage organizational IT, starting with PD research about shop floor IT management. Shop floor IT management explicates how PD tools and techniques are used to support users and professional IT developers throughout the evolving design and implementation processes. This also forms the start of how PD comes to transcend local development towards a sustained PD development approach in the organization of large-scale IS development. However, there are still many challenges of sustaining PD in the organization that are in need of more research. This thesis focuses on how one can move beyond conceptualizing the need for a PD management into practice. This raises both theoretical and methodological questions.

The following chapter puts forward an analytical framework based on "infrastructuring" to further understand how to extend shop floor IT management to a sustained participatory organizational IT management approach.

3 Infrastructures and infrastructuring

The notion of Information Infrastructures offers a possibility to conceptualize the multitude and diversity of IT applications and standards that modern organizations use in their everyday procedures. This can also relate to one specific but crucial software application as, for example, an Enterprise Resource Planning (ERP) system. Through a number of developments in organizational IT practices, infrastructures are of increasing importance also to sustaining Participatory Design in the organization. These developments include the rapid expansion of interconnected IT devices that spread across more and more applications fields and how IT support has become taken for granted in many use environments, as well as where users and organizations have become dependent on a certain quality of service delivered by IT. To this end, infrastructures can be used to highlight aspects of standardization, dependencies, and emergence from a previous technical base (Pipek & Wulf, 2009).

This chapter takes its stance from Star and Ruhdler's (1994) socio-technical notion of Information Infrastructures, which elevate a relational analysis of an otherwise common-sense definition of infrastructure as something that runs "underneath" actual structures. Information Infrastructures are then related to PD through Karasti and Syrjänen's (2004) notion of "infrastructuring." This PD approach to infrastructure development in the organization is then put in the context of predominate Information System (IS) management approaches. A critique of such approaches follows that takes a departure in Ciborra's (2000) "from control to drift." The final section develops the infrastructure challenge of this thesis on connecting different infrastructural layers of development activities, ranging from situated shop floor IT management to organizational IT management and to technical bases.

3.1 Information infrastructures

Star and Ruhleder (1994, 1996) use Information Infrastructures to target technology development that go beyond the local project and how technology affects organizational transformation. In their case, they analyzed a large-scale custom software effort, where there were challenges ranging from simple lack of resources to complex organizational and intellectual communication failures and tradeoffs. Through the use of information infrastructures, an analytical framework and vocabulary are put forward to begin to answer the question: What is the relationship between large-scale infrastructures and organizational change? (Star & Ruhleder, 1996). Star and Ruhleder's (1994, 1996) use of Information Infrastructures draws on Science and Technology Studies (STS). STS offer conceptual frameworks for analyzing Large Technical Systems, and have a long tradition of investigating infrastructures, including road and railway systems, electricity grids, and telecommunication networks. It is an interdisciplinary field of research aimed at understanding and influencing how society shapes science and technology, and how science and technology, in turn, shape society. In the analysis of Information Infrastructures, Star and Ruhdler's (1994, 1996) analytical framework deepens the socio-technical relational and situated nature of infrastructure in STS. Star and Ruhdler (1996, 1994) use Batson (1978) as a starting point to describe this extended relational property: "What can be studied is always a relational or an infinite regress of relationships. Never a thing." To analytically relate to this conceptualization of infrastructure, Bowker's (1994) notion of "infrastructural inversion" is used, "referring to a powerful figure-ground gestalt shift in studies of the development of large scale technological infrastructure" (Hughes 1983; 1989). Bowker (1994) uses the concept of infrastructural inversion to describe the fact that historical changes that are frequently ascribed to some spectacular project of an age are often more a feature of an infrastructure permitting the development of that product. This analytically emphasizes infrastructural relations, over things and people as causal factors. Based on this, the following dimensions of infrastructure are put forward:

- *Embeddedness: Infrastructure is "sunk" into, inside of, other structures, social arrangements and technologies;*
- *Transparency: Infrastructure is transparent to use, in the sense that it does not have to be reinvented each time or assembled for each task, but invisibly supports those tasks;*
- *Reach or scope: This may be either spatial or temporal -- infrastructure has reach beyond a single event or one-site practice;*
- *Learned as part of membership: The taken-for-grantedness of artifacts and organizational arrangements is a sine qua non of membership in a community of practice;*
- *Links with conventions of practice: Infrastructure both shapes and is shaped by the conventions of a community of practice;*

- *Embodiment of standards: Modified by scope and often by conflicting conventions, infrastructure takes on transparency by plugging into other infrastructures and tools in a standardized fashion.*
- *Built on an installed base: Infrastructure does not grow de novo; it wrestles with the "inertia of the installed base" and inherits strengths and limitations from that base.*
- *Becomes visible upon breakdown: The normally invisible quality of working infrastructure becomes visible when it breaks: the server is down, the bridge washes out, there is a power blackout. Even when there are back-up mechanisms or procedures, their existence further highlights the now-visible infrastructure. (Star & Ruhdler, 1996).*

The configuration of these dimensions forms "an infrastructure." The relational and situated characteristics of Information Infrastructures highlighted by Star and Ruhleder (1994, 1996) elevate the analysis of infrastructure from substrate to substance. It moves beyond a conception of infrastructure from "something upon which something else runs or operates" (Star & Ruhleder, 1994 p 252). According to Star and Ruhleder (1994, 1996) this challenges the possibility of "genuine universals"; where tasks to be automated are well-structured, the domain well-understood, and where system requirements can be determined by formal, a priori needs-assessments.

In the case of Star (ibid), the technical system building effort was an attempt to develop infrastructural tools for research. At the same time, the system development process also became an effort to bring together people from different communities of practice with different approaches to technical infrastructure. The challenge for analysis was that technology was both engine and barrier for change, both customizable and rigid; and both inside and outside organizational practices. A seeming paradox, but caused by the tension between the need for local, customized, intimate technologies, on the one hand, and the need for standards and continuity, on the other. The simultaneous need for customization and standardization is not only a challenge of technology, but also that of an organization.

The relational and situated characteristics of Information infrastructure challenges the usefulness of linear approaches to system development, as exemplified by the "waterfall life-cycle" model, still predominant in software engineering.

Traditional methodologies for system development, based on rationalistic or "mechanistic" ideas about artifacts and infrastructure, did not provide a base for understanding the infrastructure challenges at hand.

Star and Ruhleder (1994, 1996) position PD to respond to the design challenge of infrastructure, and Star and Neumann (1996) introduce Information Infrastructures in the PD community. In the context of the infrastructural design challenge of this PhD, section 3.5 also refers to Star and Bowker's (2002) succeeding work on "how to infrastructure."

3.2 Infrastructuring

Star and Ruhleder's (1994, 1994) discussion of infrastructures is further developed by Karasti and Syrjänen (2004) and Karasti and Baker (2004) with the concept of "infrastructuring", where salient features of Information Infrastructures are connected to PD. Their aim is to build on the notion of information infrastructures to understand community PD. In doing so they approach infrastructures from a bottom-up point of view compared to Star and Ruhleder (1994, 1996), who are concerned with large infrastructure projects (Karasti & Syrjänen, 2004). The term "infrastructuring" is coined to sensitize the understanding of community PD as an embedded, ongoing, and multi-relational activity, which unfolds over extended period of times. At the center of infrastructuring is the integration of new tools and technologies with existing people, materials, and tools.

In order to deepen the relational understanding of infrastructures, Karasti and Syrjänen' (2004) connect infrastructuring to Suchman's (1987, 2007) notion of artful integrations, which refers to hybrid systems comprising media, material, and practices. Design becomes a continuous process of inscribing knowledge and activities in new material forms. Artful integration also emphasizes a "located accountability" of design, where change becomes a part of everyday practice. Together, infrastructuring and artful integrations emphasize continuous and interrelated design activities that take place over time and are embedded in "multi-relational socio-material-technical contexts" (Karasti, 2014). The design of Information Infrastructures is viewed as "constantly becoming in addition to its complexly relational qualities."

In reference to PD, a connection is also made to Dittrich, Eriksén, and Hansson's (2002) "PD in the wild" introduced in the previous chapter, where PD is not only related to in terms of bringing “use to design,” that is, where professional designers are sensitized to the perspectives of end-users, but also “design in use” where end-users carry out and take responsibility for development tasks themselves. This highlights the employment of decentralized and grass-roots processes with a gradual deployment of technology closely intertwined with the development of their main work activities. It also challenges radical technological design at the “privilege” of professional designers.

Karasti and Syrjänen (2004) studied the development of software support in two Communities of Practice (CoP) with very different traits: one community of dog hoppyists and one community of information managers within a large-scale research network. The latter case is further developed in Karasti and Baker (2004). The respective community members in both these cases had in common a community identity through common causes, shared interests, and strong commitments. They engaged in a variety of tasks and practices that offered them a rich understanding of their domain. Both CoP have evolved over time as their respective members have started to experiment with design technologies alongside the ongoing development of their main work activities. Especially the importance of their informal nature, as forums for interactions to share, learn, and collaborate are highlighted. It is pointed out that through long-term relationships based on respect and trust, the members have developed a common sense of purpose and desire to share technology-related knowledge practices. Both CoP are joined by taking long-term responsibility for not only their work domain and both existing systems and procedures, but also the development of new ones.

To this end, infrastructuring becomes a way to advance overarching community interests (Karasti, 2014): “It integrated with the communities’ ongoing activities and was embedded in multiple contexts relevant for the communities over extended period of times” and allowed the communities to “grow” their infrastructures in an ongoing, long-term manner. In Karasti et al (2010), this is further described as a “continuing design” – a juxtaposition between “project time” and “infrastructure time". Applying this perspective makes visible how infrastructuring expands from “design in use” to a more inclusive approach where the boundaries between, use, design, implementation, modification, maintenance, and redesign are blurred; this “continuing design considers the past by attending to the ‘installed base and relies on temporally open-ended activity and the long-term perspective required sustainable collaborative infrastructure development'" (Karasti 2014 with reference to Karasti et al, 2010). In connection to PD, this implies an open-ended agenda giving attention to the tentative and flexible and multifarious relations and processes inherent in particular communities of PD

In the second paper by Karasti and Baker (2004), the CoP of information managers are in focus. The "infrastructural inversion" (see previous definition Star & Ruhleder, 1994) used to understand the supporting role of the information managers in creating infrastructure for collaborative ecological research is of particular interest here. It can be related to the role of the IT-professionals in the empirical material of this thesis. The Information Managers work is described as an enduring, collaborative process of infrastructuring, which is denoting the overall success of the application. At the same time, the invisibility of their work is highlighted, which is partly due to the nature of their work of providing support for ecological science, where a "busy getting the work out of the door" mentality denotes the work. Putting the infrastructure in focus enables to foreground the backstage elements of the information managers support work, such as the taken-for-granted functioning of data management and database infrastructure maintenance that by definition are part of the background. This background work is combined in their work role alongside the participation in continuous articulation work together with the users. This requires both local knowledge and working experience, and entails stringing a balance between intertwined elements and complex expertise.

In this respect, there are also methodological challenges that are recognized when studying infrastructuring, such as putting a focus on "boring infrastructure things" in addition to high-tech devices; accounting for such situated practices and large-scale collaborations, and developing methods to study long-term collaborations and their development over long periods of time.

3.3 Prevalent Information System management approaches to infrastructure development

Although Star and Ruhleder (1994, 1996) and Karasti and her collaborators (2004) are among the first to keep infrastructure development with the work practices of domain experts and IT-professionals in focus, the technical design of infrastructures has been subject to IS research before. The prevalent methodologies for system development, however, takes a top down IT governance or Enterprise Architecture

(EA) perspective, which is in opposition to the above cited “bottom up” approaches. Based on various point of entries, similar concerns about common methodologies to system development are commonly found in the PD community and in the related work presented in the previous chapter. The following section also further describes how infrastructure development, in reality, is found to be “drifting” (Ciborra, 2000), compared to common assumptions in established IS Management approaches of infrastructure.

This section describes EA based on Bernard (2005) as a concrete point of reference to manage infrastructure development, which can be related to as a today common and popular approach of IS management. The point is not that the outcome of this PhD research should (or should not) adopt or mimic such an approach to infrastructure development. However, it contributes to giving an indication of the challenges that “infrastructuring” to sustain PD in the organization is facing. It also gives a concrete reference to reoccurring assumptions of infrastructure design in concurrent IS management approaches.

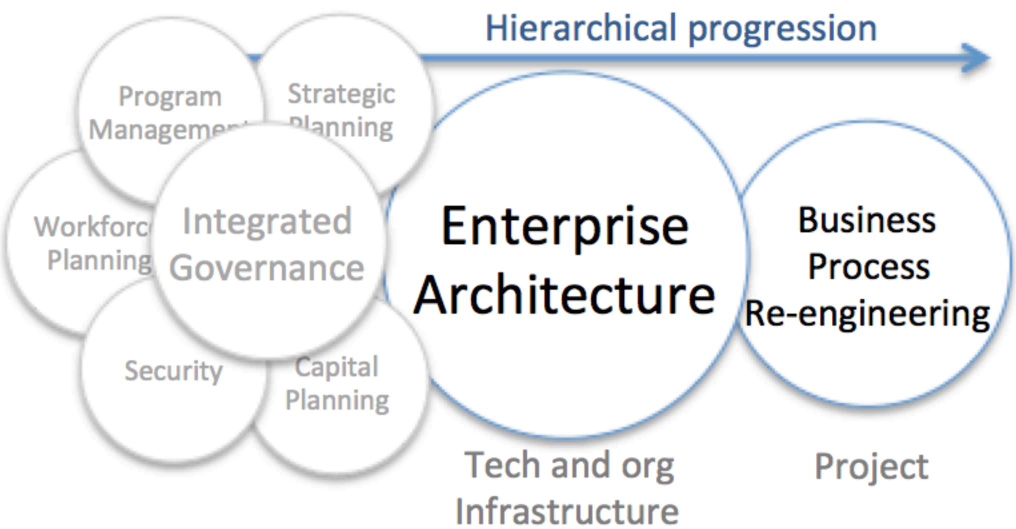


Figure 3.1. EA linked with integrated governance and project specific approaches such as Business Process Re-engineering

According to Bernard (2005), EA is driven by strategic goals and business requirements, with the intention to identify and align business and technology components of strategic initiatives. An overall issue that EA sets out to target is that technology historically has not been viewed as a strategic asset, where planning activities have often focused on the development of individual technology solutions to meet particular organizational requirements.

The idea of EA is to create abstract views of an organization (an enterprise) that helps people in the organization to make better plans and decisions, both about the organizations IT infrastructure and developing the whole organization. EA extends beyond technology planning by adding strategic planning as the primary driver of the enterprise, and business planning as the source of program and resource requirements. The place for technology planning is to meet business requirements that accomplish the strategic initiatives of the enterprise.

As a practice, EA is both a management program and a documentation method that together provides an actionable, coordinated view of an enterprise's strategic direction, business services, information flows, and resource utilization.

As a management program, EA provides a strategy and business-driven approach to policymaking and resource development that has different functions for executives, line managers, and support staff. At the executive level, EA provides visibility for large IT initiatives and supports the determination of strategic alignment. At the management level, EA supports design and configuration management decisions as well as the alignment of IT initiatives. At the staff level, EA supports decisions regarding operations maintenance, as well as the development of IT resources and services. The objective is to support a standardized approach for developing IT and other resources to reduce the risk that cost, schedule, or performance parameters are not met. In addition, to be effective, an EA program must be part of an integrated group of management policies and processes that form an overall governance structure. This governance structure includes strategic planning, EA, program management, capital planning, security, and workforce planning, as shown in Figure 3.1. On the other side of the spectrum, EA also interfaces with project specific approaches such as a Business Process Re-engineering.

As a documentation method, EA documentation is accomplished through the following six elements: (1) and EA documentation framework, and (2) an implementation methodology that supports the creation of (3) current and (4) future views of the architecture, as the development of (5) an Enterprise Management Plan to manage the enterprise's transition from the current to future architectures. The base of the documentation method is the documentation framework. Documenting current and future views of an EA helps the enterprise to identify and manage its current resources, select and implement future resources and manage the EA transition in an effective, standardized manner. The EA documentation framework identifies the scope of the architecture to be documented and establishes relationships between the architecture's areas. Bernard (2005) puts forward an example of a documentation framework called the "EA3" cube with hierarchical levels so that the different sub-architectures can be logically related to each other (see Figure 3.2). Through the way that it collects and organizes architecture information, the framework creates a complete abstracted set of "views" of an enterprise.

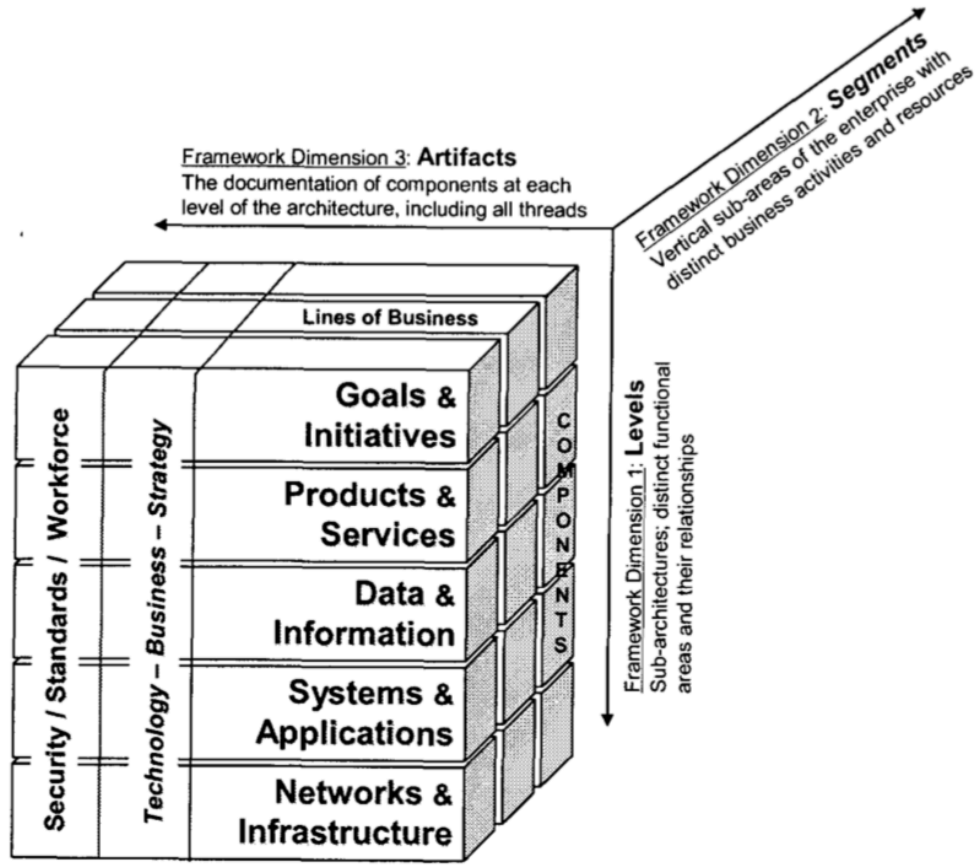


Figure 3.2 Documentation framework of EA by Bernard (2005) called the EA3 cube

The EA3 cube positions high-level strategic goals / initiatives at the top, business/products/services and data/information flows in the middle, and supporting systems / applications and technology infrastructure at the bottom. To lower the risk and promote efficient and phased implementation methods, the EA framework is divided into segments of distinct activity, referred to as Lines of Business. The documentation of EA components can be both vertical - for one line of business - and crosscutting - for many lines of business - and there is an associated catalogue of techniques both to document the current architecture and the future architecture.

EA has common denominators with other IS management approaches. According to Hedman and Kalling (2002), a common assumption is the definition of standards to make it possible to implement uniform procedures for organizational areas or functions to be shared by all actors within a defined domain, ultimately with as little inconsistency or redundancy as possible (Hedman and Kalling, 2002). To this end, the management and documentation framework EA for example intend to establish clear strategic plans to be used throughout the different levels of the organization. In this way, it is argued as being possible to develop conscious and purposeful plans on how to achieve desired results. Strategy formulation in regard to organizational development and change is concerned with planning, and often planning over a long time-horizon. This planning is at the overarching responsibility of management (Hedman & Kalling, 2002). Effectiveness criteria stem from control over the environment. Consequences of applying action strategies can be comprehended through conscious comparisons, which can then be used to further refine and structure the overall guiding strategy.

3.4 Drifting infrastructures

Andreu and Ciborra (1996) articulate a general critique against the still held wide-spread IS management assumptions described above of applying too much of a rationalistic, mechanistic, and top management centered perspective. The authors argue that the overall view on management in IT has been biased towards the analytical, the conscious, top-down, control and simplicity, structure, and the separation of action and structure. According to Andreu and Ciborra (1996), this view on the management of IT projects is incomplete and incorrect because it does not take into account the difficulty (and arguably the opportunities) with IT projects as they progress. IT projects are often nurtured and developed at operational levels in bottom-up processes, which IS management commonly fails to address. Ciborra (2000) scrutinizes a number of cases of infrastructure development in large multinational organizations and finds practices that at times “substantially diverge from the wisdom contained in the management and IS literature” of today. Ciborra (2000) pinpoints this critique with the title of the book “From control to drift- the dynamics of corporate information infrastructures.”

Simplified, Figure 3.3 describes the position put forward in relation to EA: Ciborra argues that comprehensive and integrated management and documentation frameworks that connect the wider organizational governance and local projects does not help to understand or manage infrastructure development. Instead, different situated bottom-up development processes - which can be related to Karasti and Syrjänen's (2004) and Karasti and Baker's "infrastructuring" - result in what Ciborra calls 'drift'. Although this is not of primary focus in Ciborra's (2000) review, his results appear to make him question the possibility of another type of IT management of organizational planning referred to by him as "meta-decision making forums" (Ciborra, 2000 with reference to Peppard, 1999). He asks "Why not play with the idea of a different partition between the limited scope of our management of infrastructure and the scope for the infrastructure itself to manage us?" (p 40). As is discussed in the forth-coming section, and also in the remainder of this thesis, the notion of meta-decision forums turns out to be of relevance here.

Figure 3.3 Technical and organizational infrastructure is argued to "drift" in the organization (Ciborra, 2002). This is an opposition to predominate control-oriented IS Management approaches that focus on the in-situ design of infrastructure.

In earlier contributions, Ciborra argues that such bottom-up processes in the organization are fundamental for the creation of innovative and competitive organizational capabilities. Andreu and Ciborra (1996) use Argyris and Schöns' (1978) development of single- and double-loop learning in an organizational context.

The learning loop processes in Figure 3.4 connect the development of capabilities in the organization to the work practices of people working on the shop floor and how they have different resources, such as software support, available to them. Using the notions of bricolage and radical learning, Andreu and Ciborra (1996) conceptualize how single- respectively double-loop learning takes place through bottom-up processes in the organization; from the development of local work practices to strategic core capabilities. Bricolage is based on tinkering characterized by shop floor people as combining and applying known tools and routines at hand to solve problems. Radical learning, on the other hand, attacks a potential competency gap at its roots. It involves empowering users to radically learn new things by becoming aware of the current context and explicitly stepping out to present different, and sometimes conflicting, perspectives for each other in order to innovate in a new manner (Andreu & Ciborra, 1996) The appropriation of single- and double-loop learning as bricolage and radical learning connects design to innovation in the organization. Andreu and Ciborra (1996) are, for example, used by Hedman and Kalling (2002) to conceptualize alternative bottom-up business models to support innovation in the organization.

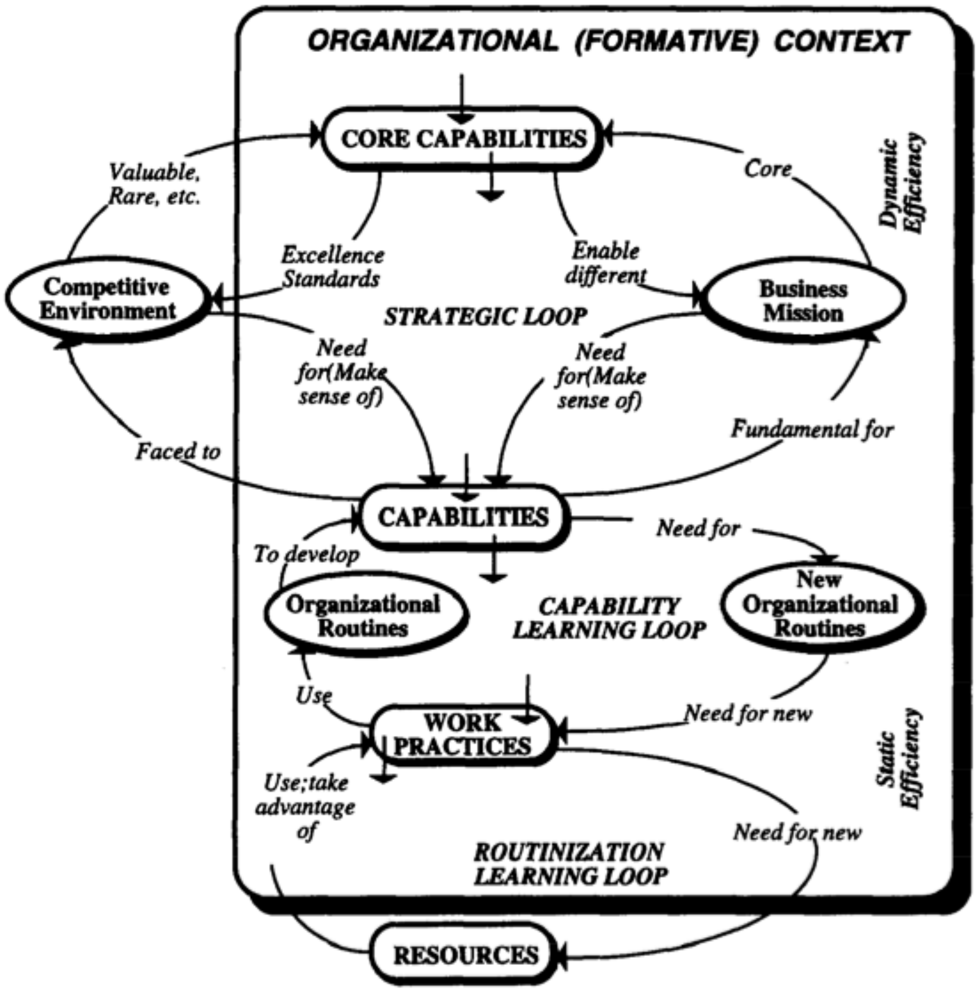


Figure 3.4. Learning in capabilities and core capabilities development process from Andreu and Ciborra (1996).

Other researchers take up Ciborra's concepts and report on empirical cases of the "drifting" nature of technical and organizational infrastructure that goes against the logic of predominate IS management approaches. Hanseth and Braa (2001) target the strategic construction of a standard and its succeeding organizational implementation through an IS infrastructure. They use the metaphor "trying to catch the treasure at the end of the rainbow" to describe the endeavor: Infrastructure standards turn out to be not as universal as assumed; instead, they are only universal as abstract constructions. When the standards are implemented, they are linked to and integrated with local technical systems and working practices resulting either in work arounds or local appropriations. In 2009, five articles in a special issue of the Journal of the Association for Information Systems were devoted to new kinds of Information Infrastructure and e-infrastructure studies. How new infrastructure in different ways must integrate with an installed base that includes not only artifacts but also human habits, norms, and roles are a re-occurring theme in these articles.

Ribes and Finholt (2009) report about four scientific cyber infrastructure projects in the US, where, on the one hand, the tension between customizable and standardized infrastructure plays out in the design and testing of novel and their risky systems, on the other hand, the viability of long-standing domain science programs with requirements of robust infrastructure could be threatened.

Sahay, Monteiro, and Aanestad (2009) study an effort to link to previously unconnected health IS in India. Continuous integration, they find, is crucial to evolving infrastructures. However, the initial choices as to which people should be involved and which technologies should be used create long-term implications for the unfolding and success of integration ventures.

Ure, Procter, Lin, Hartswood, Anderson, and Lloyd (2009) document the tensions and challenges confronting efforts to share data across and within disease domains in a UK healthcare project. As argued by the authors, these tensions can rarely be isolated and resolved at a purely technical level and with one-size-fits-all standardized infrastructure. Instead, winning strategies are most likely to be found in the relation between technical and organizational practices.

Broadly parallel stories are found first in Hepsø, Monteiro, and Rolland (2009) in the efforts to deploy a new Microsoft Sharepoint infrastructure. The specific tension is "between implicit and explicit top-down demands for tight integration embedded in the SharePoint infrastructure and how these unfold dynamically against the persistent, bottom-up reliance on niche systems and micro-practices commensurability." Second, Pipek and Wulf (2009) use the notion of infrastructuring (Karasti & Syrjänen, 2004) in the analysis of the results from a long-term study of IS and work processes developed to link legislative processes in a German state government with those at the federal level. They introduce the concept of "work infrastructure" to direct analysis only towards which IS users actually use and how they use them, rather than the full range of facilities available to them. Pipek and Wulf's (2009) work infrastructure framework is of particular relevance to this thesis as it is used to conceptualize PD and infrastructuring in different layers of infrastructure and technology development (further developed in the following section). In addition, Pipek and Wulf (2009) connect infrastructuring to end-user development, which, in turn, can be related to the shop floor IT management approach discussed earlier. In line with Erikséns (1998) concept, Pipek and Wulf (2009) see that infrastructuring transcends the traditional distinction between IT-professionals and users.

There are also other contributions where the information infrastructure concept is used to further the analysis of areas such as Computer Supported Cooperative Work (CSCW). Monteiro, Pollock, Hanseth, and Williams (2012) propose to re-articulate an open-agenda of design in CSCW referring to information infrastructures. Design in CSCW, they argue, needs to incorporate the non-local constraints of standardization - "how local fitting entails unfitting at other sites" – and embeddedness - "the entanglement of one technology with other apparently unrelated ones."

3.5 The "infrastructuring" challenge to sustain Participatory Design in the organization

When sustaining PD in the organization, the challenge is to connect different layers of infrastructure development, with a particular focus of how situated IT management on the shop floor relates to organizational IT management dimensions, and the need to coordinate technical base development. This pushed research in this study to move beyond the conception of technical and organizational as "drifting", and look additionally into the type of "meta-decision-making forums" that Ciborra (2000) appears to question.

Pipek and Wulf (2009) develop the concepts of "work infrastructure" in a framework of infrastructural layers of technology development activities (figure 3.5) relate infrastructural background work and implementation of the infrastructure to in-situ design work like tailoring and appropriation of the infrastructure.

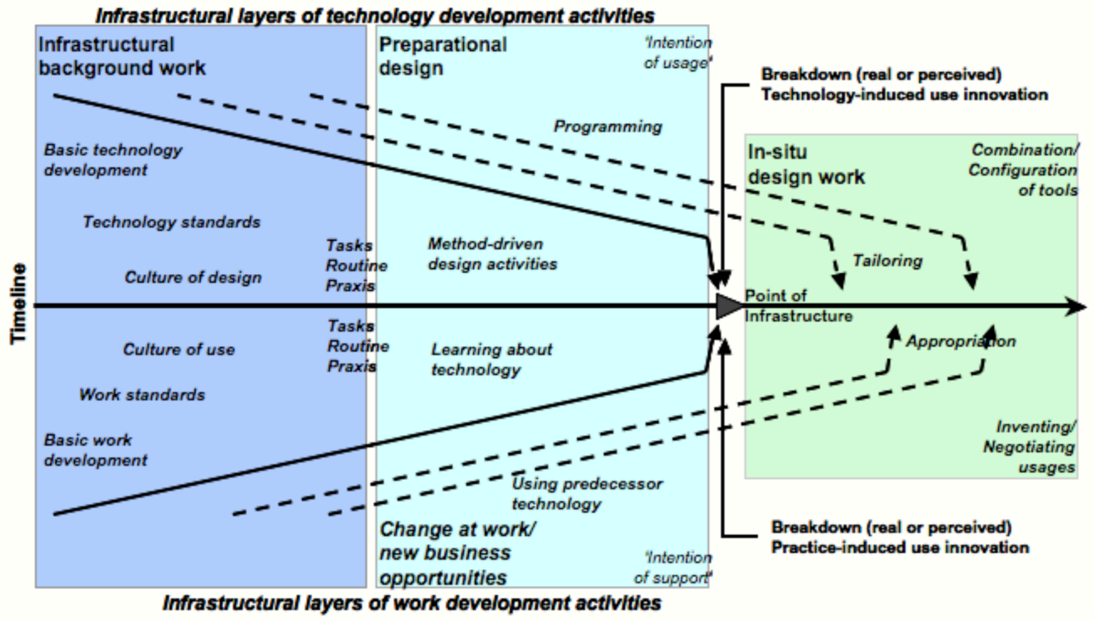


Figure 3.5. Pipek and Wulf's (2009) infrastructural layers of technology development activities

In Pipek and Wolf's (2009) framework, infrastructuring takes its stance in "points of infrastructure." Figure 3.5 indicates the point of infrastructure defined as when the routines of performing work meet the technology development activities of professional designers in a work / use or technology breakdown that then triggers new innovation. From then on (right side), the local development of infrastructure configuration and usage are considered as in situ-design or design-in-use as opposed to design-before-use (Pipek and Wolf, 2009, with reference to Pipek and Syrjänen, 2009). At these points of infrastructure, normally invisible elements of the work infrastructure may become salient for actors present in the situation, regardless of whether they are professionally trained or not. This moment catalyzes both informal and formal "in-situ design work" by designated designers and users who reconfigure and / or extend the existing work infrastructure to repair the breakdown. Pipek and Wulf (2009) point out that any actual work infrastructure includes numerous user innovations, and that the traditional conception of designers rarely if ever is able to take full account of the complete systems and practices involved in local accomplishing work goals. It is argued how a wide variety of work practices – tasks, routines, and praxis – prepare both users and professional designers for "points of infrastructure" design. These are triggered by temporary breakdowns in a work infrastructure's actual or perceived ability to provide services. A second point of infrastructure then is “use innovation” when users successfully appropriate a new infrastructure for a local context.

Even before this moment (middle of Figure 3.5) there are relevant activities going on for some actors already engaged in developing or considering infrastructure both in regard to use / work and technical design. These activities are called “preparatory design work.” On both the technology and use side, Pipek and Wolf (2009) refer to these activities as indicating an intention to support work infrastructure.

Finally, even before this (to the right of Figure 3.5), other relevant activities occur, which are called “infrastructural background work.” These activities can again take place either in the technology development or work sphere and are more strategic in nature, informed by issues that may have emerged from previous points of infrastructure. This includes basic technology and work development, as well as the development of work and technology standards. It may also include meta-design activities, which Pipek and Wulf (2009) exemplify with users who seek additional qualifications (such as programming skills) in order to be more efficient at improving their own infrastructure.

In the action research reported in this thesis, the challenge to connect to “infrastructural background work” (Pipek &Wolf, 2009) when designing for and with different shop floor development constituencies triggered the development of structures, processes and techniques to include local design constituencies into the design of the infrastructure. These activities needed to be addressed explicitly and related to the local design. Putting focus on such technical and organizational meta-infrastructure activities is needed when providing an alternative to traditional IS management methodologies.

This is in line with Star and Bowker's (2002) book chapter ‘How to Infrastructure’– even though they address large-scale infrastructure development beyond an individual organizational context. Star and Bowker (2002) argue that background work needs to be understood if we are to produce thoughtful analyses of infrastructure. They also recognize the possibility of drawing on PD when responding to the design challenge of infrastructure development. As Star and Bowker (2002) note, “The infrastructure designer must always be aware of the multiple sets of contexts her work impinges on. Frequently a technical innovation must be accompanied by an organizational innovation in order to work: the design of sociotechnical systems engages both the technologist and the organization theorist.” The authors also point out that this dimension has received comparably little attention: “Both standardization and classification are essential to the development of working infrastructures” at the same time “work done on standard committees and in setting up classification schemes is frequently overlooked in social and political analyses of infrastructures, and yet it is of crucial importance” (Star & Bowker 2002, p 154). They analyze different strategies for standard setting, where at one end of the spectrum, the strategy aims at one standard fits all; at the other end, there is the “let a thousand standards bloom model.”

Star and Bowker (2002, p 156) put forward a metaphor that becomes interesting when transferred to an organizational context: they call the former a “colonial” model of infrastructure development where the latter is the “democratic” model. The challenge for managing design is to allow for flexibility, and still allowing for example information to persist over time. This is recognized not an easy task. Star and Bowker (2002) describes how the required flexibility, in general, is emergent, where “it is clear that you do not need a single great architect for an infrastructure” (p. 159). Instead, these social arrangements “are best developed and maintained by standard bodies containing representatives of all major stakeholders” (p. 159).

In this thesis, the challenge of the ‘colonial’ vs ‘democratic’ model is to understand what it means in an organizational setting, both in terms of structures and procedures and processes. In greater detail, the challenge becomes to create a connection between the in-situ development of different shop floor development constituencies and organizational IT management, a connection that is based on extending users shop floor IT management to the organizational arena. This can be related to the PD topics described in the previous chapter underpinning a sustained PD approach in the organization. Put in the context of Björgvinsson, Ehn, and Hillgren's (2010) "agonistic" approach, the challenge becomes to empower a multitude of voices from the shop floor in organizational IT management.

3.5.1 The technical base in infrastructuring

In addition to the organizational IT management dimension of infrastructure development, the technical base becomes in focus in this thesis. The technical base comes to the forefront as planning device in the process of “infrastructuring” in order to understand the impact of technical design processes in shop floor development constituencies. As Star points out (2002, p.154), “it is not just the bits and bytes that get hustled into standard form in order for the technical infrastructure to work. People's discursive and work practices get hustled into standard form as well.” From a design point of view, it is possible to analyze how the installed base of a particular infrastructure changes over time;, how it sometimes carries inertia, but then sometimes transforms remarkably rapidly, sometimes apparently discontinuously.

Compared to Karasti and Baker (2004) and Karasti and Syrjänen (2004), the traits and influences of the technical base underpinning an infrastructure in themselves become an important denominator for the flexibility of development and spaces of design in the individual development project. In this sense, the “from somewhere” of Suchman’s (1994b) located accountability includes also the technical base. It is part of the existing technology and work practices to which design needs to be related. As Orlikowski (2010) notes, while research that views technology as a process situates the production and use of technology in particular socio-cultural and historical contexts, “it also tends to downplay specific technological properties and affordances, focusing primarily on human interpretations and social actions” (p. 11).

Leonardi (2010), for example, argues for a focus on imbrications of not only human, but also material agencies that evolve infrastructure in the form of routines and technologies that people use to carry out their work. The technical base in regard to infrastructure development is also present in the 2009 special issue (Volume 10, Issue 5) of the Journal of the Association for Information Systems (JAIS). Howcroft and Light (2010), Hepsø, Monteiro, and Rolland (2009), and Sahay, Monteiro, and Aanestad (2009) purport that technologies have different characteristics affecting development, such as how functionality and modules are packaged and interlinked with each other in one specific implementation project. By giving attention to the technical base in “infrastructuring,” this study adds to the state of the art by focusing on the interaction between the technical base and participatory and sustainable evolution with respect to heterogeneous and developing requirements. In chapter 7 it will become visible how the technical base plays into the long-term interaction between IT-professionals, users, and EUDs over several generations of technical infrastructure development.

3.6. Summary

Through the notions of Information Infrastructures and infrastructuring, this chapter has conceptualized an alternative to organizationally managing IT infrastructure development in a participatory way. This approach is an alternative to predominate Information System Management approaches, as exemplified through EA. It also provides a way forward to research contributions that recognize the "drifting" nature of infrastructure development. The situated nature of infrastructure development is still in focus, but is complemented with shop floor IT management extending to the organizational arena.

The need for a new conceptualization for a participatory management of infrastructure development in the organization emerged in-situ through the course of the empirical research. In the following chapters, the research approach and the empirical findings are described. This is followed by a combined discussion that relates the contributions to the related work in this and the previous chapter of infrastructuring for sustained PD in organizations.

4. Research approach

To answer how end-users can participate in the evolution of an organization's IT-infrastructures prompted a long-term action research engagement.

An opportunity that the empirical domain of this PhD research offered was an existing long tradition of established shop floor IT management practices. End-users in different shop floor development constituencies at the World Maritime University (WMU) were accustomed to manage the development of their software support. In supporting the development of technical and organizational infrastructure using a Participatory Design (PD) approach, it was important to continue to work especially with these end-users on the shop floor, to develop their capabilities from their perspective. As the author of this thesis, it was an opportunity to combine employment as an IT-professional with embedded research in order to understand, deliberate, and evaluate improvements of infrastructuring to sustain PD in the organization. Carrying out research in this way, however, presented both an opportunity as well as a challenging route to conduct research. Both these dimensions are accounted for in the sections below.

The question is, however, whether it could have been done in another way? The main part of the empirical research came from action research where I worked together with colleagues at the WMU to develop software, both to support internal socio-technical process improvements and enhance the university's curriculum offering. This put certain requirements on the research approach: it was required to account for long-term evolutionary design and implementation over multiple locations, and to be able to take a diverse set of contingencies into account. From a research approach point of view, design activities occurred not only as part of planned project activities, such as scheduled meetings and design workshops, but were equally situated in day-to-day encounters – in other words "use in design" as well as "design in use." The research approach thus had to be able to extend beyond a local project setting, a common focus of PD research (for an example of early influential PD projects see Bjerknes, Ehn, Kyng, & Nygaard, 1987; for a contemporary example see Bødker, Kensing, & Simonsen, 2004).

This chapter describes how Dittrich, Rönkkö, Eriksson, and Hansson's (2008) Cooperative Method Development (CMD) approach as a guiding methodological framework was appropriated for the purposes of this PhD research. CMD is a cyclic and iterative research approach that combines deliberation through action research with supporting ethnographic research to understand and evaluate changes made in the empirical domain. The objective of action research is to influence or change some aspect of the empirical domain to learn more about a certain course of events (Robson, 2002); in this case, the changes addressed how end-users could participate in the evolution of an organization's IT infrastructures. In PD, as part of a Scandinavian collaborative research practice and tradition, action research has been a popular approach for researchers to expand scientific knowledge, while at the same time solving practical problems (Baskerville & Myers, 2004).

The first part of this chapter describes how the CMD approach is appropriated and applied in this research; how CMD is extended to include users as designers to address infrastructure development; how ethnographically inspired empirical research supports the action research to understand and evaluate changes made in the empirical domain; how shop floor development and use practices are put in the center; how the shop floor perspective is put to work in the research, that is, in taking the practitioner's perspective when evaluating the empirical research and deliberating improvements; and finally how deliberations of improvements are carried out together with the involved practitioners. The second part of this chapter provides a breakdown with a timeline of how the CMD approach was applied.

4.1 Cooperative Method Development as appropriated and applied

For the empirical research in this thesis, the ethnographically inspired CMD approach by Dittrich et al. (2008) is used. From the outset, CMD was chosen because a structured methodological framework was called for to support the embedded research nature of the empirical research that target both "use in design" in planned project activities but also continuous "design in use." CMD is based on Checkland and Holwell's (1998) cyclic process guideline for action research, moving between observation, planning of the action to be taken, and implementation of the action (see Checkland & Holwell, 1998 for a comprehensive presentation of the action research CMD). It also relates to Mathiassens (1998, 2002) Reflective Systems Development as a structured way to evaluate development methods; CMD was originally developed to improve the inclusion of cooperative aspects in Software Engineering research. It has been applied and refined over a number of research projects that address different parts within Software Engineering: design of flexible and adaptable software

(Diestelkamp & Lundberg 2000); use-oriented development (Dittrich & Lindeberg 2003); agile development for e-government applications (Hansson, Dittrich, & Randall 2004; Hansson, Dittrich, Gustafsson, & Zarnak 2006); and the integration of interaction design and software development (Rönkkö, Hellman, Kilander, & Dittrich, 2004; Rönkkö & Dittrich, 2005).

In this thesis, CMD is appropriated beyond Software Engineering to address (1) PD and a focus also on the development of the use organization and (2) to include technical and organizational infrastructure from the user's perspective. As Pipek and Wulf (2009) also point out, infrastructure improvements may or may not involve technological reconfigurations or the introduction of new tools. This development is further described in the sections below and discussed in chapter 10.

The structure of CMD is defined by five guidelines (Dittrich et al., 2008):

1. An action research cycle consisting of three phases: understanding, deliberating change, implementation, and evaluation of improvements
2. Ethnographical inspired research complemented by other methods if suitable
3. A focus on shop floor software development practices
4. Taking the practitioner's perspective when evaluating the empirical research and deliberating improvements
5. Deliberating improvement with involved practitioners

These five dimensions of CMD and how they relate to the research in this thesis are further described in the following sections.

The appropriation and use of CMD in this thesis can be related to a call for improved methodological support for action research both to address sustained PD in the organization (Simonsen & Hertzum, 2012) as well as in research on infrastructuring (Pipek & Wulf, 2009):

- In their sustained PD approach, as introduced in the related work chapters, Simonsen and Hertzum (2012) extend an iterative approach to PD, where evaluation of systems in real situated work practices is emphasized. A PD experiment is presented where four phases of an iterative research are put forward: identifying desired change; specification and implementation; real use enabling unanticipated change; evaluation. Applying it in the organization, however, is recognized to raise a number of methodological challenges in need of more research: creating appropriate conditions for PD, managing a multitude of stakeholders; managing a stepwise implementation process; and conducting realistic large-scale PD experiments. They argue that these “how-to” challenges cannot be satisfactorily answered with general methodological guidelines. What is called for instead is more research, preferably action research that refines the PD approach by applying it in a number of cases and thereby stimulating the mutual creation and sharing of knowledge and experiences (Simonsen & Hertzum, 2012, p. 16). The CMD sets a clear focus on shop floor development activities of users and answers the challenges put forward by Simonsen and Hertzum (2012). From the outset, in addressing the challenges of creating the appropriate conditions for PD when managing a multitude of stakeholders, the core focus on shop floor users that denotes the above guidelines is valuable.

- Also in regard to related work on infrastructuring, CMD answers methodological challenges put forward. In classical IS methodologies of infrastructure, core design questions as “When does the interaction take place?” and “What is the kind and depth of information exchanged?” are methodologically answered by professional IT designers (Pipek & Wulf, 2009, p. 460). They define when they need information and what information they need. Pipek & Wulf's (2009) work infrastructuring framework highlights a needed users' perspective to this by: *methodologically advising users to perform frequent procedures aimed at infrastructure improvement that may or may not involve technological reconfigurations or the introduction of new tools; providing methods, as well as tools, to systematically perform these procedures; and preparing and engaging in interactions with the traditional professional design sphere*. To this end, CMD provides an approach where users and IT-professionals can engage in infrastructuring. As is described in the following section, CMD contributes a structured approach by putting the user’s perspective in the center. In this thesis, this is recognized of particular value when working with complex layers of infrastructure development, which are expanding from the situated realities of the shop floor.

4.1.1 The Cooperative Method Development research cycle

In this thesis, the research process with the CMD approach is modeled as evolutionary research cycles of qualitative empirical research of technical and methodological innovations. These are carried out and evaluated in cooperation with the involved practitioners. The application of CMD is defined as a domain-specific adaptation of action research consisting of three phases, which can be repeatedly applied in the same context in accordance with Figure 4.1 below. In this thesis, a PD dimension of organizational and associated methodological innovation is added to the original software engineering focus of CMD. In addition, the inclusion of an infrastructure dimension extends the action research in CMD beyond the local project. In the research process of this thesis, it means that action research was carried out in different and intertwined layers of infrastructuring. The three phases of the action research in CMD are introduced below, and then further defined in the coming sections. How action research with CMD was carried out is also described with a timeline in section 4.3.

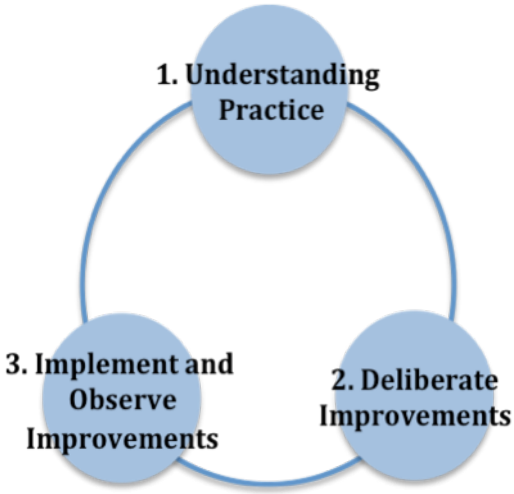


Figure 4.1 Three phases of the CMD framework

- Phase 1 - *Understanding practice:* the CMD research cycle begins with qualitative empirical investigations into the problem domain. Through the use of ethnographical and ethno methodological inspired research, the aim is to understand and explain practices and designs from practitioner's' point of view. The intention is to understand existing practices in their historical and situational context, and to identify aspects that are problematic from the involved practitioners' perspective. Also, in line with the intent of CMD, the engagement with and analysis of the empirical results unfold in a "grounded theory fashion" (Dttrich et al., 2008, p. 6; also see Robson, 2002, p. 190). Section 4.1.3 and section 4.1.4 account for the bottom-up research process of this thesis, where research contributions are not the result of an "a priori" theoretical framework, but are grounded in an analysis of issues in the problem domain. Understanding of practice took place both during participatory observation of every day encounters (I was physically placed in the middle of the faculty landscape and had overlapping technical, administrative, and academic work duties) and during planned participatory workshop observations of particular issues. The use of ethnography, ethnomethodology, and grounded theory is further described in section 4.1.2.

- Phase 2 – *Deliberation of improvements:* The results of the first phase are used as input for the deliberation of possible improvements, covering, in this thesis, the technical, organizational, as well as methodological improvements in different infrastructural layers of technology development (also see Pipek & Wulf, 2009, described in the related work chapters). These deliberations of improvements are done in cooperation with user and IT-professional practitioners involved in the infrastructure development. The result of this phase is the deliberation of measures that can be expected to improve the situation at hand and address jointly identified problems. Deliberations of improvements took place through both planned project activities and everyday design interactions. For the former, empirical material was collected in the form of audio recordings in combination with project documentation and documentation of tools and techniques used; for the latter, research diary notes were primarily used for documentation, together with ad-hoc audio recordings for personal use (a further important dimension was to carry out retrospective evaluation with the people involved).
- Phase 3 – *Implementing and observing improvements.* During the implementation of the improvements the researcher follows these method improvements as participant observer. The results are evaluated together with the involved practitioners. The results of the evaluation help to summarize concrete results for the organization involved. The results of the evaluation are also used for researchers involved to build the base for the scientists' evaluation of the proposed improvement measures. In the context of this thesis, the evaluation of implemented improvements was primarily done

through retrospective semi-structured interviews and workshops with people involved.

The empirical research features three CMD research cycles with different appropriations of the CMD phases in practice. An overview of these is given in the timeline in section 4.3, which are then further detailed in the following chapters. To enable the reader to understand how the empirical research unfolded, these chapters are structured in accordance with the three phases of the CMD research cycle.

4.1.2 Ethnographically inspired empirical research

Similar to the approach taken in Dittrich et al. (2008), focusing on software development practices encouraged the use of qualitative methods for the empirical work. For this purpose, the CMD approach uses ethnography and ethnomethodology as a theoretical underpinning, both to understand existing practices and to evaluate changes implemented as proof of the action research. The use of ethnography is an important component in CMD to focus analysis on the shop floor development and use practices in the deliberation of design. In this way, the research outcome of CMD is not a standalone ethnography, but ethnography forms a base to deliberate and account for design choices. The use of ethnomethodology additionally addresses the methods a social group uses to organize cooperation and communication.

Robson (2002) describes an ethnography as providing a description and interpretation of the culture and social structure of a social group. It often involves immersion in the particular culture of the social group studied so that life in that community can be described from the members' view point. Being able to gain an insider's perspective is seen as a desirable feature. Participatory observation in the field is considered essential, although no method of data collection is ruled out in principle. Reporting on ethnography research calls for a detailed description, analysis, and interpretation of the culture-sharing group. This account is typically written in a narrative literary style. In the context of reporting on IS research, Dittrich et al. (2004) notes that such rich descriptions are sometimes misunderstood to be unscientific story telling. However, they are intended to allow other scientists to follow and where necessary, argue conclusions of the research. Apart from the evaluation of the field material, the design of the research must also be presented and argued.

Immersion in the empirical domain and the capability of providing an inside account were facilitated in this research in that the author of this study was working as an embedded researcher. During this research, I was employed both as a PhD researcher by the IT-University of Copenhagen and as an IT-Professional by WMU. Working as an embedded researcher enabled the empirical research to go beyond planned participatory observations. This is recognized as an important complement to acquire an understanding of what Suchman (2007, p. 16) discusses as the “just here, just now” achievement of social order based on the situated action of people. In this thesis, this, for example, relates to shop floor IT management and end-user development. In addition, activities of importance to different layers of technical and organizational infrastructure development were going on both inside and outside planned research and organizational activities. An episode of importance to changes to the technical base (described in chapter 7) illustrates this point; during an everyday encounter between faculty assistant Levy and myself noted in my research diary, he explained why he was not using a newly developed academic scheduler: “You were so pre-occupied with your technical stuff […] you were just not listening to me […] that was not going to work.”

This episode is described in detail in chapter 7 in the context of what triggered the need to change the technical base in the action research of the first CMD research cycle. It is part of a complex socio-technical infrastructure relation of how a technical base interplays with shop IT management, and also indicates how a technical base can both enable and constrain PD. It was also an admittedly difficult episode for myself. The academic scheduler was one of my first major development undertakings. It was ordered by the vice-president (academic), who was also one of the initial organizational sponsors of this PhD.

Capturing the dynamics of this important episode of infrastructure development was enabled by the embedded nature of this PhD research, where I was immersed with a long-term engagement in the application domain. Agar (1996, pp. 135-139) describes the importance of immersion and long-term engagement with an empirical domain in ethnography as enabling one to reflect and be critical of the social circles where the research takes place. He exemplifies this with a note that early project sponsors and facilitators can be “deviants” that have something to gain by establishing a relationship with the researcher per se. In the case above, if I had come from the outside as part of a planned research activity, with no established relation to the organizational stakeholders, it would have been difficult to capture the dynamics of the infrastructure development. Chances are that given the Vice-President’s keen interest in the development as well as support of the research, faculty assistant Levy would not have been so up front about his concerns to me. In this way, the deeper engagement with the application domain that the embedded action research enabled was important to get a more multifaceted account of the development dynamics, which included Levy's perspective.

The episode also highlights different techniques of how ethnographic field material data were recorded and documented. These can be divided in two parts: planned and in-situ techniques. Planned ethnographic techniques included participatory observations and interviews. The main in-situ technique was the keeping of a research diary. The research diary was used to document the day-to-day events and interactions with organizational actors, meetings, and workshops. It was also used for organizational and project documentation. To this end, the research diary served the purpose of informing both the ethnographical research and the action research. In addition, ongoing empirical material collection was made up by project documentation and analysis of programming code (the latter to understand changes of how technical development patterns evolved in developed modules).

Given the ongoing and embedded nature of the research, the in-situ research diary was a primary documentation technique, whereas planned participatory observations and interviews often were supportive and used for triangulation. In the above episode, Levy’s in-situ statements were, for example, triangulated both through other research diary entries of other users’ experiences, as well as it being confirmed later in an follow-up interview. To triangulate empirical data collection, a number of retrospective workshops and interviews were conducted with the key people in their roles as users, end-user developers, IT-professionals, and managers (often these roles were combined with a given person). These additional empirical data collection techniques are described in the forthcoming sections

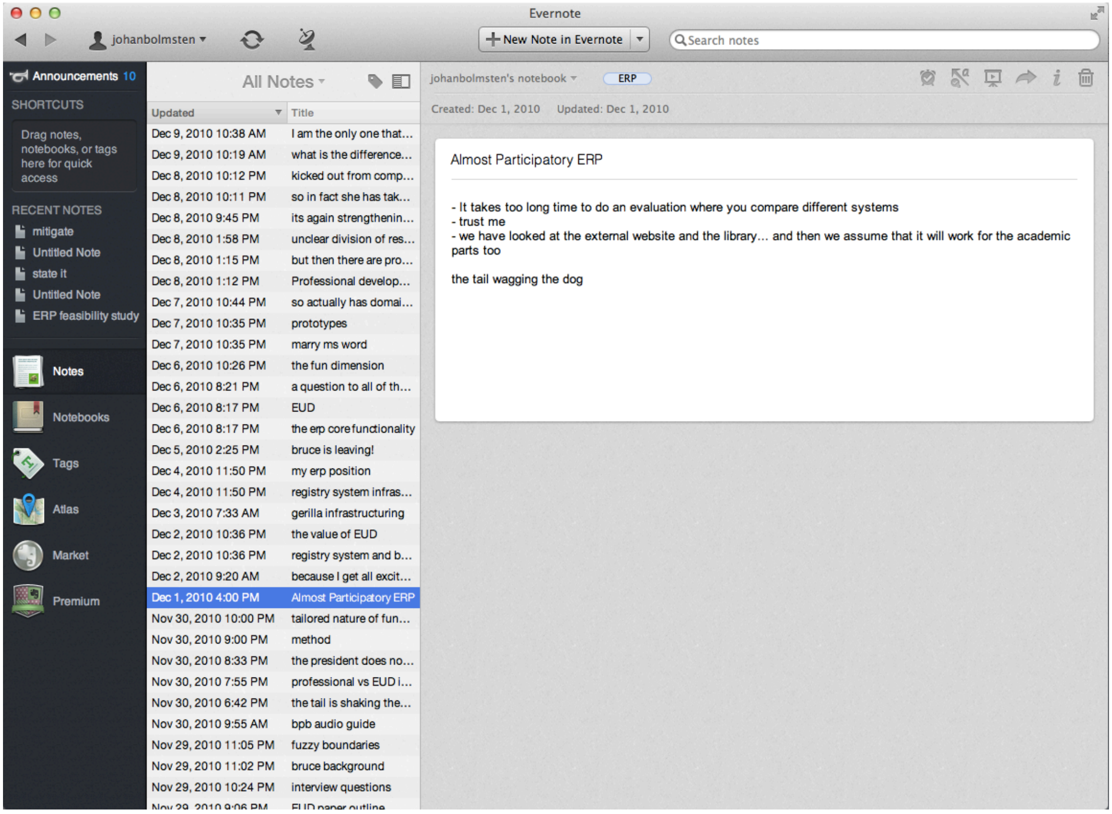


Figure 4.2. The software used for the research diary was called Evernote. It is a cloud-based tool that is available across several platforms, including laptop, tablet, and smartphone, as well as through a web-based interface.

To be able to capture day-to-day events, the research diary was set up to be ready-at-hand and support several input formats, including text, audio entries, and pictures. After completing the empirical work, the research diary contained 1,500+ entries. In addition, there were, in total, 132 audio recordings that were made of development interactions during informal and formal meetings, workshops, and participatory observations. The software used was available across several different platforms (see Figure 4.2). Research diary entries could be inputted from applications installed on a laptop, smartphone, tablet, as well as from a web-browser interface. The laptop was the primary unit used. However, frequently short text notes or audio recordings of in-situ events were also made on the smartphone (In the case that audio recordings were used beyond the purposes of memory notes, the explicit consent of the recorded person/s was acquired.). Entries made on one platform were automatically synchronized into a central repository available from all platforms. In addition, the research diary software enabled a categorization of notes that could be used as an input for later analysis (see section 4.2)

Externally, the supervising professor Dittrich has continuously supported the research through offsite debriefing sessions (and conducting interviews). This was one response to how the embedded nature of the empirical research not only provided opportunities but also challenges, such as personal biases and stakes in the development projects, as I was employed by the organization. Section 4.4 at the end of this chapter summarizes working with the trustworthiness of the research.

4.1.3 Focusing on shop floor development and use practices

An important motivation for using the CMD approach in this PhD research was the explicit focus on shop floor IT management practices. Dittrich et al. (2008) also state that the motivation to develop the CMD approach in the first place was a desire to understand how software developers tackle the everyday challenges of developing useful software. This thesis supports a way of expanding this Software Engineering focus on software developers with a PD focus also on use organization and an End-User Development, where the users in part even take on technical development of the infrastructure themselves.

An opportunity for the empirical research about how end-users can participate in the evolution of an organization's IT infrastructures was the already established long-tradition of shop floor IT management at WMU. The value of shop floor IT management was already recognized on an organizational IT management level. For example, the IT-steering committee described in chapter 9 was an early and bottom-up initiated organizational IT management effort.

In this way, the focus on shop floor IT management practices should therefore not be mistaken for ad-hoc behavior. As Dittrich et al. (2008) also note, the term "practice" describes an established way of doing things, where the practice is produced and re-produced through the action of those who take part in the practice. In this way, the individual's action is visible and understandable for his or her peers as meaningful behavior with respect to the common frame of reference that shared practices provide. In Dittrich's et al. (2008) case the focus is on developing software. Here this is extended with a focus on PD and infrastructure development. As also described in the related work chapter through the focus on "practice" of shop floor IT management, the explicit starting point becomes how participatory IT management process, procedures, and structures on the organizational arena can work as "resources for action" (Suchman, 1987; 2007). The starting point is thereby not generalizable methodological organizational IT management frameworks as found in classic Information System research, but to ground solutions in the work practices and situated realities of users on the shop floor and IT-professionals.

All the empirical research and the description thereof that are accounted for in the following chapters are grounded in an understanding of different shop floor IT management practices. A narrative vignette is provided in the beginning of each chapter. For the purposes of critical reflection, such account makes it visible how the situation under investigation emerged, and sets the scene of the research. In reference to the breakdown of the application of the CMD approach in the following section 4.3, the number of shop floor constituencies and technical and organizational infrastructures gradually expanded as the research progressed. They both expanded in terms of the number of people involved in a given shop floor development constituency and the number of shop floor development constituencies per se. This also provided opportunities for triangulation to improve the trustworthiness of the research (see description in section 4.4).

4.1.4 Taking the practitioners' perspective when evaluating the empirical research and deliberating improvements

Even when focusing on shop floor IT management, "the perspective under which the observations are evaluated and that guides the choice of improvements can be one of management or users" (Dittrich et, al, 2008). The importance of being clear about the perspective applied in the phases of the CMD research cycle is noted both in the cases presented in Dittrich et al. (2008), as well as in the background work (see, for example, Dittrich, 2002).

In this way, in the episode about what triggered the technical infrastructure development, for example, it was important to assume from faculty assistant Levy's perspective of the issues that he faced. The Vice-President (Academic) was a supporter of the shop floor IT management approach at WMU, and sponsor of the PhD of how to develop this approach on the organizational IT management arena. He had also initiated the technical development of the first academic scheduler himself as a teaching staff, before becoming a manager. However, it was important to ground the evaluation of the empirical research and deliberations of improvements in the situated reality of the practitioner currently carrying out the main work on the technical infrastructure. Similar attention to the practitioners' perspective was applied in the other CMD research cycles of this thesis as well. The development of the registry system was, for example, carried out by Registrar Davis himself, who was also a supporter of shop floor IT management. However, his account of the development was complemented with interviews and participatory observations with the Registry Assistant Magnusson and Student Services Officer Evans, all of whom were important everyday shop floor users of the system.

Another conscious choice that was critical to be able to take a practitioner's perspective when evaluating the empirical research and deliberating improvements was my position in the organization. The PhD research included participatory IT management in the organizational arena. However, I was not the manager carrying the research. At the beginning of this research, I had the status of a supporting staff and at the time of finishing the research, I had an officer position. Both are lower level positions. If I would have been a manager, and, for example, faculty assistant Levy had reported to me, then the dynamics in our relation would have been different, as probably also the position he would have put forward on development, and consequently my understanding of what he was saying. This does not mean that I did not have to be conscious and continuously reflect on my own perspective, but it provided a beneficial vantage point.

One central technique that was developed to triangulate my own perspective and gain an overview of the collected empirical material in the research process was to invite practitioners to participate in timeline workshops. The timelines were developed and applied in two different formats during the research (their position and use in the research process is further mapped and described in section 4.3). The first timeline was developed at the end of the first CMD research cycle, together with practitioners during two succeeding workshops. Development events were mapped on a white board, as shown in Figure 4.3. After the workshops, the timeline remained on display for informal reflection and input also by other stakeholders.

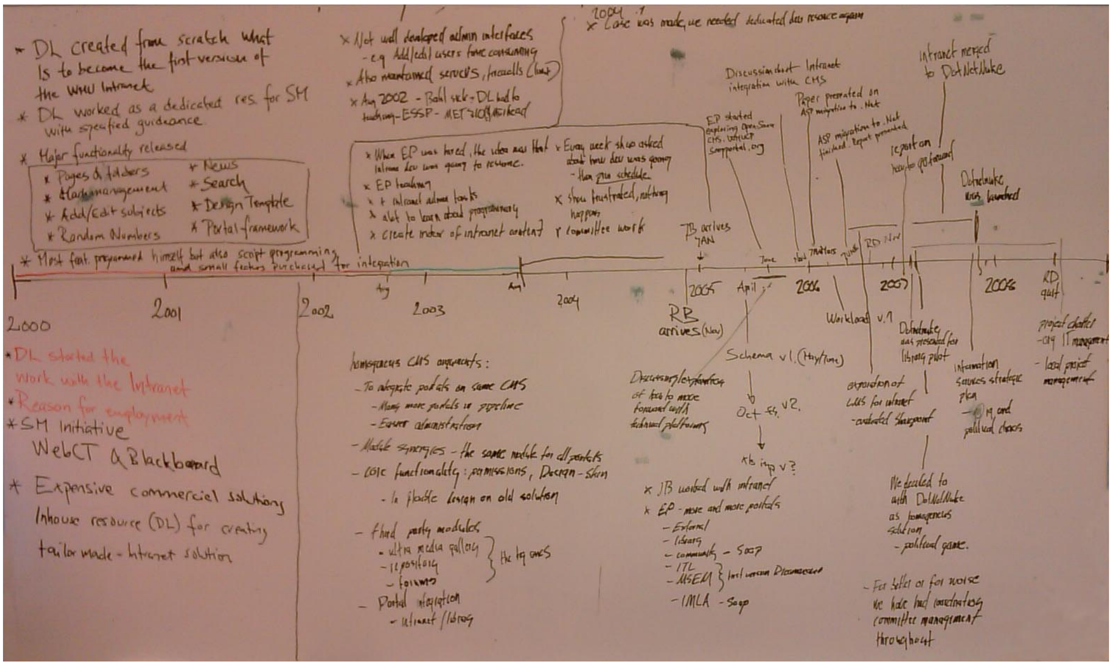


Figure 4.3. Outcome on whiteboard from timeline workshop

At the end of the third and final CMD research cycle, the second timeline workshop was constructed, using a different format. Using the mapping tool MindManager, gathered empirical material such as project documentation, calendar and research diary entries, audio recordings, images of implemented interfaces, prototypes, and mock-ups were reconstructed in a timeline format (see Figure 4.4). In total, 600+ instances of the gathered field material were mapped. The complete timeline was then printed and posted on a wall (see Figure 4.5) and used as a reference during a number of interviews. Similar to the first timeline, it was also used for informal reflection after the interviews.

Figure 4.4. Timeline mapped using MindManager.

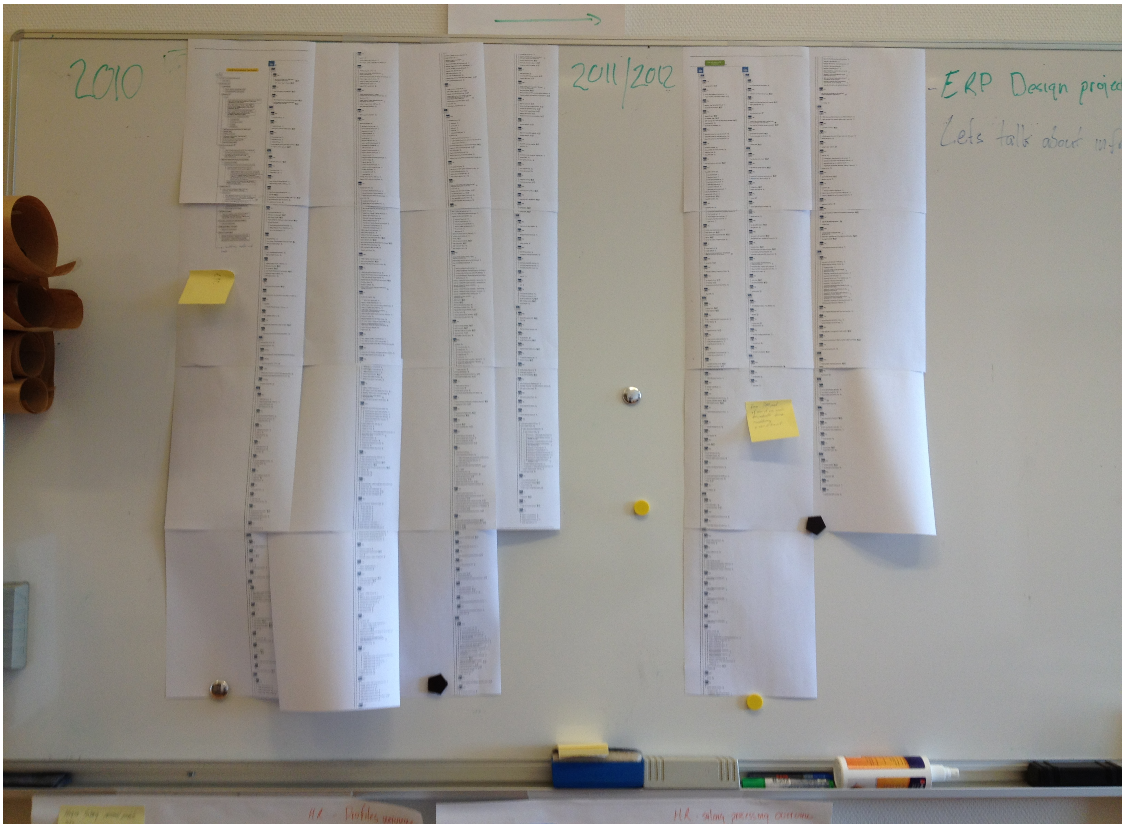


Figure 4.5. Timeline on whiteboard to among else support reflections during follow up interview with Magnusson and Evans

4.1.5 Deliberating improvements together with the involved practitioners

It can also be recognized to have jeopardized the trustworthiness of the research if I could have ordered the development of what then would have been my version of a participatory IT management. During the research, I had the power to suggest improvements in various settings and forums as part of the action research, but no formal coercive power to enforce them. For this purpose, it was an opportunity to take advantage of the rich set of methods, tools, and techniques proposed by the PD community, addressing the combined design or work and software support (See Bødker et al. 2004).

Three primary internal arenas were used to involve different constellations of shop floor users, IT-professionals, and managers in deliberating improvements: (1) Situated in the middle of the faculty landscape, development discussions would typically take place in offices of different stakeholders, myself included, and deal with matters of relevance of ongoing projects. These meetings would often be informal in nature, but could also be planned and structured by a set agenda. Meetings took place both pre and post developments to be able to discuss implications of decisions made. Collages and sketches on white-boards, notes, mock-ups made in PowerPoint and Photoshop, and horizontal and vertical prototypes are examples of artifacts for collaboration used. Other primary arenas for ongoing reflection were the meetings of (2) the IT-coordination group and (3) the IT-steering committee. These meetings were both used for informal reflection – much similar to the above description of the local project context – and for more structured deliberations and prepared workshops.

A primary challenge facing the action research deliberations was how to bring embedded matters of technical and organizational infrastructure development to the surface for PD. As one early structured means for joint reflection in the CMD research cycles, a series of what came to be called “reflection papers” were developed for both action and research purposes (referenced in the case story). They were written based on the idea of story cards (Beck & Andres, 2004) and intended to take their stance in issues facing concrete shop floor developments of software support as a common base for reflections of organizational and technical considerations of the choice of technical bases. Emphasis was put on writing the reflection papers in a style and language that were meaningful to users working on the shop floor as well as IT-professionals. Together with other boundary objects, such as prototypes, they became a central part in both the action research deliberations and the development approach in itself. In the interactions with people in the shop floor development constituencies, they were both used for informal reflection and for more structured deliberations and prepared workshops. They were also used in the communication with the IT-steering committee:

- In total, three "reflection papers" were produced as part of the first CMD research cycle and were used to understand the technical base as a socio-technical relation; to discuss how a new software architecture could be utilized from a "use in design" perspective; and to deliberate on the final evolvement of the technical base to enable "use in design."
- In the second CMD research cycle, the reflection papers from the first CMD research cycle were carried over and amended in the appropriation of MUST as a management and knowledge framework (Bødker et al, 2004) to deliberate an improved organizational IT management
- In the third CMD research cycle, the reflection papers were systematically connected to other PD tools and techniques. In their final version, the reflection papers were written based on participatory observations and workshops by this researcher during the process of understanding of work practices. They were then iterated with the end-users until a final version could be confirmed before being presented.

4.2 Analyzing and accounting for the field material

For each research contribution, after an initial analysis, a selection of research documentation and audio recordings of particular relevance were transcribed. These were then analyzed with the qualitative research tool, HyperResearch. The analysis started with identifying codes in the transcribed material. Based on this open coding, a number of categories were developed which were used for axial coding, relating the different transcripts (see Figures 4.6 and 4.7). Code maps were also developed to cluster different categories in relation to each other. Externally, Professor Dittrich joined the evaluation of the interview transcriptions and the identification of reflection themes (see Figure 4.8).

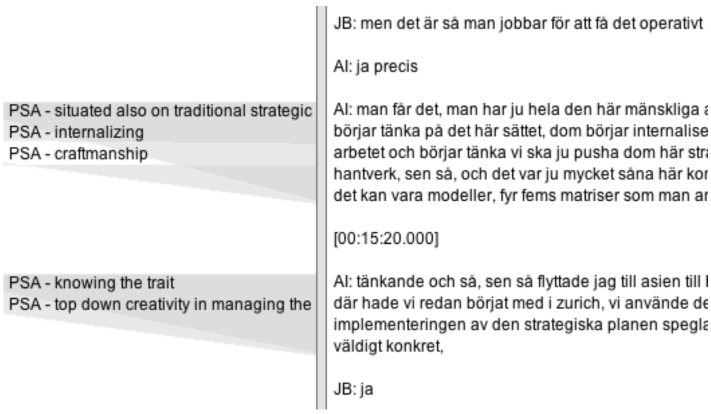


Figure 4.6. Codes in the context of the empirical material

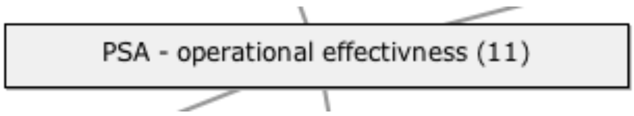


Figure 4.7. A code category, indicating how many times it is applied in the empirical material

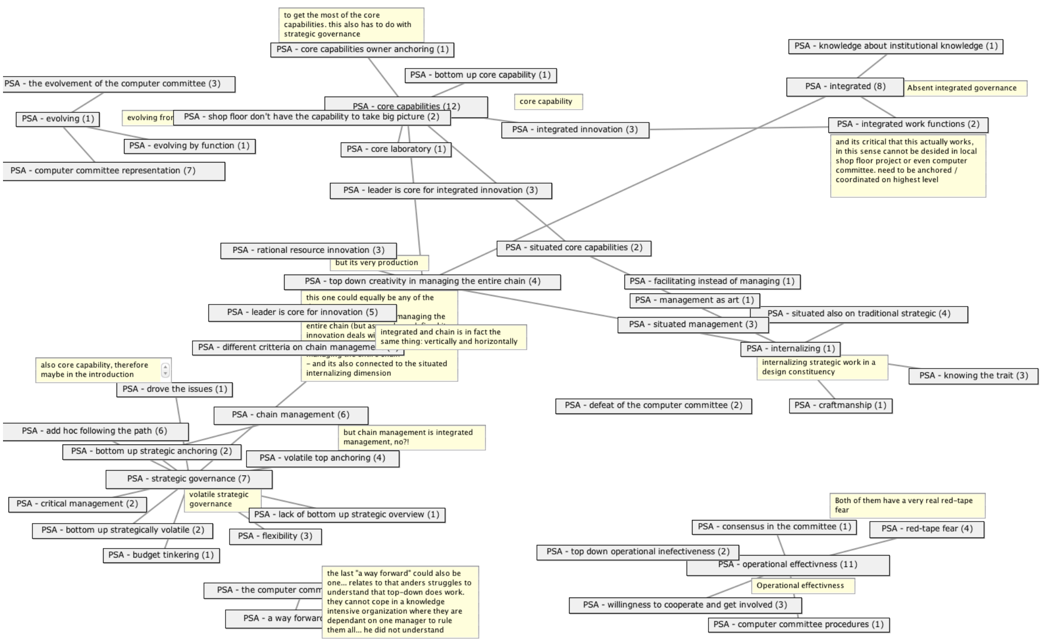


Figure 4.8. A code map clustering and relating code categories with notes

In the data analysis, Agar's (1996) notion of rich points have been used for theoretical guidance. According to Agar (1996), rich points are what "fuels" ethnographic research. They are defined as those surprises and departures from the researchers' expectations that signal a difference between the researcher and the culture of the group or phenomena under study. This is pointed out as a necessary process of translation between the researcher and what is researched, which set a perspective on ethnographic research that is both relational and partial:

- Relational: "Whenever we hear the term culture, we need to ask, of whom and for whom" (Agar, 2008, p. 5). In this sense, culture is not a property of them, nor is it a property of us. It is an ethnographic construction built to enable translation between them and us (Agar, 2008, p. 6)
- Partial: At the same time, culture is partial. Culture is always plural. We are always part of multiple cultures that might or might not mix in given situations, based on, for example, dominance of a culture, how coupling occurs between one culture and others, and how much relevance a certain cultural has for a particular situation (Agar, 2008, p. 7).

Culture as a translation moves away from closed, coherent systems of meaning and action in which an individual always and only participates. The use of rich points and viewing culture as a translation enables two types of accounts to report research:

- Encyclopedic shared knowledge ethnography: What can be thought of as the classic encyclopedic "shared knowledge" ethnography, Agar (1996, pp. 12-16) refers to as a "disk" contribution. The goal is to find common threads in the sense of "patterns" or "value configurations" that range across several cases where the local and individual complications wash out. In the above episode with faculty assistant Levy, it was, for example, possible to find other related occurrences where end-users had key influence of the development of infrastructure.
- Narrative ethnography: A narrative ethnography, on the other hand, can be referred to as a "pick" contribution (Agar, 1996). It accounts for the practices of everyday life, the way those practices are built out of shared knowledge, plus all the other things that are relevant to the moment. It foregrounds chasing rich points across domains and levels, rather than chasing them in order to translate a specific culture in use (Agar, 2008, p.

9); in this case, in a Community of Practice or Community of Interest by some person/activity to coordinate their social world. This gives the ability to catch individual "complications and contradictions, not as evidence for the encyclopedia, but as problems to explain in their own right" (Agar, 1996). The account of how faculty assistant Levy's objection to the new academic scheduler came made a difference in different layers of infrastructure development in the organization, as accounted for in chapter 7, exemplify a "pick" contribution in the analysis of the empirical material.

In this thesis, these two types of ethnography have been used in combination to understand and evaluate the empirical material. Following Agar's (1996) propositions, this research does not treat these two approaches of analyzing and reporting on ethnographic research as mutually exclusive; instead, they are used in combination. A similar approach is used in, for example, Dittrich and Lindberg (2002) in the context of IS research. A narrative account that keeps close to the details of an everyday account is combined with looking for common patterns in the empirical material. The latter can either serve as a foreground or background to the former, but the idea is not to "wash out" individual complications when looking for common patterns across cases (Agar, 1996, p. 10). The following chapter 5, 6, 7, 8 and 9 that report on the empirical research of this thesis all combine these two types of ethnographic account. They first give a narrative account of the empirical domain of study, and in chapter 7, 8, and 9 of the action research carried out. Then follows a summary of experiences that are generalized for further discussion in chapter 10.

4.2.1 The voice of the research

Choosing an ethnographic research style has consequences not only for how an empirical domain is methodologically approached, but also for how the research is reported. In ethnographic research, the personal pronouns "I" and "we" are used to indicate that studies have a particular voice and are written from a particular point of view. This form of writing is a consequence of the ethnographic emphasis on fieldwork experience, as opposed to fieldwork findings, which is commonly associated with the reporting style in software engineering (Rönkkö, 2005). Being clear about "the voice" viewpoints and motives of the researcher are of particular importance in this thesis, where the researcher is closely associated with the empirical domain.

4.3 Breakdown of Cooperative Method Development application

This section gives a breakdown of how the CMD research cycle was put to work. The major part of this chapter accounts for how the action and ethnographic research unfolded to enable the reader to understand how the empirical research was conducted and how it was interlinked. In the following chapters, different dimensions of the empirical research and its outcome are presented in greater detail.

The tables in this section provide a reference for readers so that they can return to them to peruse where the different actors and infrastructure elements were involved. The first tables account for the different shop floor development constituencies that were part of the empirical research. These are the primary units of analysis in this research study. The second table indicates which technical platform infrastructures were used by the shop floor development constituencies. The third table describes the organizational IT management infrastructures that members from the shop floor development constituencies were engaged in to manage the technical platform infrastructures. Finally, the fourth table introduces the people that were involved in the action research deliberations and evaluation. The source and driver of both situated action and planning in approaching the management of sustained PD in the organization is the shop floor development constituencies at WMU. The technical platform and organizational IT management infrastructures are recognized as supporting planning structures.

4.3.1 Shop floor use and development constituencies

The primary unit of the empirical research is a number of shop floor development constituencies. These are native in the sense that they existed before the research commenced. Their composition, size, and development scope changed throughout the research.

Faculty academic administration	Users, IT-professionals, and managers make up the shop floor development constituency of the faculty academic administration. The faculty academic administration is developed in-house and contains software support such as mark management, scheduling, e-documents, and syllabus management.
Registry academic administration	The shop floor development constituency of the registry academic administration contains users (also with a management capacity) that also work as End-User Developers. The registry academic administration is developed in-house and contains software support such as enrollment, student records, and mark management.
Administration	The shop floor development constituency of the administration primarily consists of users that also worked as end-user developers. The administrative support that is part of the research is developed in-house and contains software support such as electronic-forms and an address database.
Finance / HR	During the research finance and HR functions were combined at the university. The finance and HR uses standardized off the shelf software support. End-user development by the users mainly concerned basic appropriation and configuration.
Additional shop floor development constituencies	Shop floor development constituencies around WMU's external website, library, research project portals, alumni portal, and dedicated technical support to extend the university's curriculum offerings, such as simulation and e-learning, are additionally featured in the empirical material.

Table 4.1. Shop floor development constituencies

4.3.2 Technical platform infrastructures

Two integrated technical bases are subject to technical infrastructure development

Web technical base	Technically integrating faculty course administration, WMU's external website, research project portals, and alumni portals.
ERP technical base	Technically integrating faculty course administration, registry, administration, finance / HR, and research project management.

Table 4.2. Technical platform infrastructures

4.3.3 Organizational IT infrastructures

Computer committee / IT-steering committee	WMU's main structure for organizational IT management. Contains user representatives from the different shop floor development constituencies, IT-professionals, and management representatives. In reflecting on an upgraded function to take on organizational IT management matters, the computer committee was renamed to the IT-steering committee as a result of the deliberations in chapter 9.
IT-coordination group	This group coordinates the work of the IT-professionals, who are formally under the management of local shop floor development constituencies.

Table 4.3. Organizational IT infrastructures

4.3.4 People involved

All the people introduced by name are involved in action research deliberations and in ethnographic research regarding understanding practice and evaluation of improvements. People involved are referenced with anonymous surnames. Common surnames are used that reflect their national origin. The introduction below is extended in the context of the individual chapters.

Position, Name, and Nationality	**Involved in (1) Shop floor development constituencies (2) technical platform infrastructure (3) organizational IT management infrastructures**	**Background**
Head of Administration ***Andersson***, Swedish	(1) Administration, finance, and HR (2) Integrated ERP (3) Chair of IT-coordination, Administration	United Nations agency, Banking
Vice-President (Academic), ***Bouchard***, Canadian	(1) Faculty course administration and research project management (2) integrated Web and ERP (3) computer committee member and IT-coordination group chair	Senior government civil servant, Head of Canadian government agency
Registrar, ***Davis***, United States	(1) Registry (also technical developer of registry system) (2) integrated ERP (3) computer committee member	Registrar at United States university, NGO in Africa
Student Services Officer, ***Evans***, Australia	(1) Registry (2) integrated ERP	Employed in various positions at WMU since mid 1980’s

Finance Assistant, **Garcia**, Philippines	(1) Finance (2) integrated ERP	University degree in finance from Lund University
Lecturer, ***Hansson***, Swedish	(1) Faculty course administration (2) integrated Web (2nd version)	Captain at container / freighter vessel
Specialization Professor, ***Hughes***, Australia (retired)	(1) Faculty course administration, e-learning and simulation	Captain at container / freighter vessel
Academic Dean, ***Laine***, Finland	(1) Faculty course administration, research project management (2) integrated Web and ERP	Lawyer, commune civil servant, Judge
Faculty Assistant, ***Levy***, British	(1) Faculty course administration (2) integrated Web	Administrator in London police office, Graphical designer for London firm
IT-professional, ***Larsson***, Swedish	(1) External website, research portals, library, and alumni (2) Integrated Web and ERP (3) computer committee and IT-coordination group member	BSc in Information Systems at Blekinge Institute of Technology, Part of Siemens training program in Beijing, China.
Registry Assistant, ***Magnusson***, Swedish	(1) Registry (2) Integrated ERP	Administrator at large multinational
Specialization Professor ***Mercier***, French	(1) Faculty course administration (2) Integrated Web	Associate Professor of Maritime Economics and Finance of Euro Management, the Business School of Marseilles, Director of the Institute of Shipping and Trade at the University of Nantes
Specialization	(1) Faculty course	Professor in Japan and

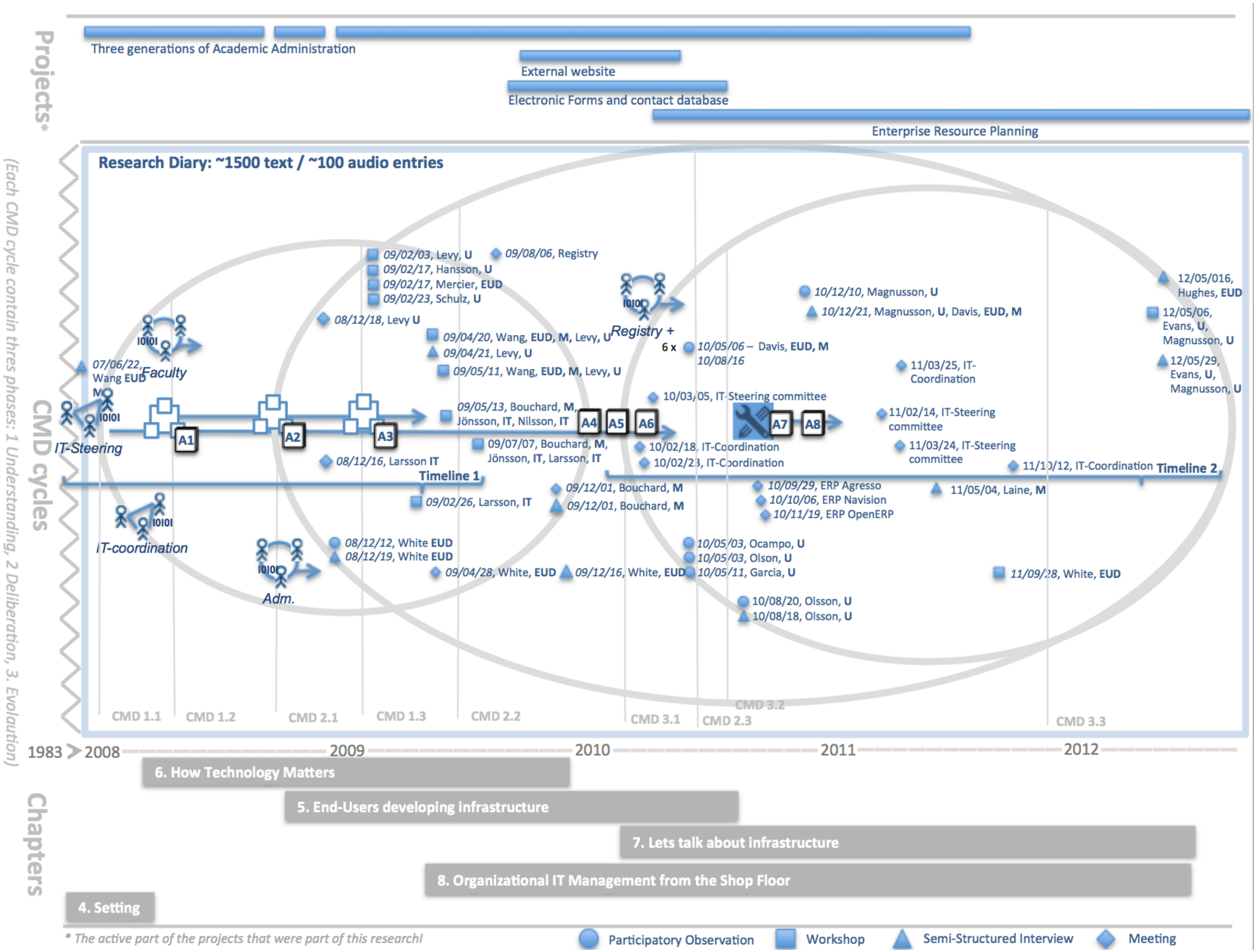
Professor, ***Nakamura***, Japanese	administration (2) Integrated Web (3) computer committee member	Canada
IT-professional ***Nilsson***, Swedish	(1) Faculty course administration	Student Malmö Högskola
Finance Officer, ***Ocampo***, Philippines	(1) Finance (2) integrated ERP	Financial Officer at UN agency
Human Resource Officer, ***Olsson***, Swedish	(1) HR (2) integrated ERP	Employed at WMU since mid 1990's
Specialization Professor ***Schulz***, German	(1) Faculty course administration and research project management (2) integrated Web and ERP	Navigation officer
Network Administrator ***Jönsson***, Swedish	(3) Computer committee and IT-coordination group member	Network Administrator Malmö university
Librarian and Head of Information ***Moore***, United States	(1) Library (3) computer committee member and IT-coordination group chair	Librarian and information system development in united states
Vice-President (Academic) ***Wang***, Chinese	(1) Faculty course administration (2) integrated Web and ERP	Un agency in Switzerland
Administrative Assistant ***White***, United Kingdom	(1) Administration (2) integrated Web and ERP (3) Computer committee member	Employed at WMU since mid 1980's
Other people involved: In addition to the people directly referenced, most WMU staff members at some point appear in the empirical material. This is also the case for a number of students. These staff members and students have made important contributions in the course of different development ventures and in the triangulation		

of empirical data. However, as they have not explicitly participated in ethnographic research regarding the evaluation of improvements, they are kept anonymous.

Table 4.4. People and their involvement in shop floor development constituencies, and the development of technical and organizational infrastructures.

4.3.5 Timeline

The remainder of this chapter puts forward a timeline and overview explanation of how the three CMD research cycles unfolded. This is put in context of the development projects that were part of the empirical research and the chapter that details the empirical research contributions in the following chapters. The timeline further shows how the shop floor development constituencies, the technical platforms, and organizational IT management infrastructures introduced have come together in the research process. In this sense, it is also shows how the research for the individual contributions has worked to triangulate each other. The research allows for gradually being able to iterate between different shop floor development constituencies and technical and organizational infrastructure entities. This was a fundamental component of building up an understanding of the empirical domain and deliberations.



Projects*
Three generations of Academic Administration
External website
Electronic Forms and contact database
Enterprise Resource Planning
CMD cycles
(Each CMD cycle contain thres phases: 1 Understanding, 2 Deliberation, 3. Evolaution)
Research Diary: ~1500 text / ~100 audio entries
07/06/22, Wang EUD
Faculty
IT-Steering
IT-coordination
Adm.
Registry +
6 x
A1
A2
A3
A4
A5
A6
A7
A8
Timeline 1
Timeline 2
09/02/03, Levy, U
09/02/17, Hansson, U
09/02/17, Mercier, EUD
09/02/23, Schulz, U
09/08/06, Registry
08/12/18, Levy U
09/04/20, Wang, EUD, M, Levy, U
09/04/21, Levy, U
09/05/11, Wang, EUD, M, Levy, U
09/05/13, Bouchard, M, Jönsson, IT, Nilsson, IT
09/07/07, Bouchard, M, Jönsson, IT, Larsson, IT
08/12/16, Larsson IT
09/02/26, Larsson, IT
09/12/01, Bouchard, M
09/12/01, Bouchard, M
08/12/12, White EUD
08/12/19, White EUD
09/04/28, White, EUD
09/12/16, White, EUD
10/05/06 – 10/08/16 Davis, EUD, M
10/03/05, IT-Steering committee
10/02/18, IT-Coordination
10/02/23, IT-Coordination
10/09/29, ERP Agresso
10/10/06, ERP Navision
10/11/19, ERP OpenERP
10/05/03, Ocampo, U
10/05/03, Olson, U
10/05/11, Garcia, U
10/08/20, Olsson, U
10/08/18, Olsson, U
10/12/10, Magnusson, U
10/12/21, Magnusson, U, Davis, EUD, M
11/03/25, IT-Coordination
11/02/14, IT-Steering committee
11/03/24, IT-Steering committee
11/10/12, IT-Coordination
11/05/04, Laine, M
11/09/28, White, EUD
12/05/016, Hughes, EUD
12/05/06, Evans, U, Magnusson, U
12/05/29, Evans, U, Magnusson, U
CMD 1.1
CMD 1.2
CMD 2.1
CMD 1.3
CMD 2.2
CMD 3.1
CMD 2.3
CMD 3.2
CMD 3.3
1983
2008
2009
2010
2011
2012
Chapters
6. How Technology Matters
5. End-Users developing infrastructure
7. Lets talk about infrastructure
8. Organizational IT Management from the Shop Floor
4. Setting
* The active part of the projects that were part of this researchl
Participatory Observation
Workshop
Semi-Structured Interview
Meeting

Figure 4.9 Indicative timeline of projects, CMD research cycles, and chapters

The timeline in Figure 4.9 gives an indicative overview of how the empirical research has unfolded. Three parts are related along the horizontal timeline that shows the research from its formal initiation in 2008 to the end in the middle of 2012. In the top, the primary development projects of software support that were used in the research are shown. In the middle, the three CMD research cycles are shown. Finally, in the bottom it is shown how the following chapters account for the empirical research.

The middle of the timeline shows the three CMD research cycles and how they are interlinked through the three phases: 1) The understanding of existing practices in their historical and situational context; 2) Action research deliberations of improved technical and organizational infrastructure; 3) Evaluation of implemented improvements. The middle section of the timeline also shows where the shop floor development constituencies and the technical and organizational infrastructure introduced earlier enter the empirical research. In addition, an overview is given of empirical data collection points that have been selected to be of primary relevance and have been analyzed to support the documentation in the research diary. The research diary with its about 1500+ text and 100+ audio entries frames the empirical research and serves as a foundation to document the empirical research (as is described in the following chapter, certain events of relevance taking place before the formal initiation of the research are also incorporated in the research). The symbols under the timeline indicate the different types of empirical data collection: participatory observations, workshops, semi-structured interviews, and meetings. In mapping the empirical data collection points in the timeline, the date, the surname of the person or persons involved, and their role (User, IT-Professional, End-User Developer, and Manager) are also displayed. Academic empirical data collection in the faculty and registry are mapped in the top. In the middle, the empirical data collection in regard to infrastructure development is mapped. In the bottom, the administrative empirical data collection is mapped. In Appendix 1 a table is provided with a breakdown of all the empirical data collection points mapped.

As an overview to the detailed account of the empirical research in the following chapters, a short description below shows how the three parts of the timeline – projects, empirical research, and chapters - link together and how the research in the CMD research cycles unfolded. It should be noted that the use of the CMD research cycles for action research responded to needs in regard to improved technical and organizational infrastructure that had their origin in practice. The three CMD research cycles, as shown in the timeline, were planned and unfolded in-situ – not predefined by the researcher before the research started. They were also prompted by each other, and their interlinking supported cross-evaluation.

The first part of the timeline indicates that matters of relevance to the research took place before the CMD of this thesis was initiated. As described earlier, shop floor IT management was an established occurrence in the organization, which de-facto was an important opportunity for the research. This is described in chapter 5. To this end, the IT-steering committee, as an infrastructure to organizationally manage shop floor development, can be traced back to 1983. The IT-steering committee is therefore mapped in the timeline before the initiation of the first CMD research cycle. The arrow then indicates how the IT-Steering Committee was part of both the deliberations of the first and second CMD research cycle. The arrow also shows where the IT-Steering Committee was complemented with the IT-coordination group, and that it connects to the deliberations of the technical infrastructure.

The first CMD research cycle about How Technology Matters in participatory infrastructure development is accounted for in chapter 7. The first phase of the CMD research cycle was triggered by a need for improved technical infrastructure (CMD 1 phase 1). It took its stance when this researcher was hired as a faculty IT assistant to work dedicatedly with the further development of the course administration software support at WMU. This initially implied a narrow perspective, where the focus was on the close dynamics of the faculty shop floor IT management. The issue that the action research targeted during the second phase of the first CMD research cycle (CMD 1 phase 2) was how a technical base could simultaneously enable and constrain the PD of the domain experts, and how there were synergies to be gained through a shared technical base with other shop floor development constituencies. The action research deliberations entailed a rapid expansion in regard to the technical base. As indicated in the timeline, three succeeding parts of action research were used to deliberate the technical base: A1) from raising the awareness of the technical to the surface and being able to talk about it, A2) to improving the capabilities of working with users in design, and A3) finally revising the technical aspects to enable "design in use." The main body of retrospective evaluation of implemented improvements was enabled through a series of timeline workshops (CMD 1 phase 3). As is described in section 4.2, timelines were used in the evaluation of the empirical material both in structured workshops as well as on public display for informal discussions and feedback. The first timeline was created together with Larsson (IT-Professional) and then used in a follow-up workshop with Wang (Manager and End-User Developer) and Levy (User).

Chapter 6 is about shop floor users also taking on infrastructure development, outside the realm of infrastructure development activities managed by IT-professionals. Beginning with, and prompted by, the technical base also gave an opportunity to engage with the two additional shop floor development constituencies of the registry and administration to understand the "native" shop floor development approach at WMU. These were historically based on end-user development. The research positions itself as an ethnographic case study as part of the first and second CMD research cycle where the aim was to further the understanding of the End-User Development activities in the context of infrastructure development. The empirical material was collected from a number of the participatory observations, interviews, and workshops with Administrative Assistant White (End-User Development), Registrar Davis (Manager and End-User Development), and Registry Assistant Magnusson (User). The empirical material analyzed here was collected to understand End-User Development practices from the End-User developers' point of view.

Chapter 9 accounts for the second CMD research cycle about Organizational IT management from the shop floor. The second CMD research cycle was comparably the longest and most complex. It connected to both the first and third CMD research cycle. The shared technical base in the first CMD research cycle prompted an understanding of issues relating to a need for an organizational IT management from the shop floor development constituencies' point of view (CMD 2 phase 1). In addition, the respective shop floor IT managements of both the electronic forms / address database and the registry system provided an alternative view that did not traditionally rely on IT-professionals, but that needed to be part of an organizational IT management. As is reflected in the timeline, as the CMD grew in scope so did also the need and number of empirical data collection points to support the documentation of the embedded research in the research diary. The reflection papers that deliberated the technical base in the first CMD research cycle were again used here as an early input to understand the issue of organizational IT management. The second and third reflection papers concerned both technical and organizational infrastructure matters. They intended to prompt reflection both to improve interaction between the newly formed IT-coordination group and how it interacted with key shop floor users. To this end, a number of presentations and discussions were carried out with the intention to purposefully use the already existing IT-steering committee for organizational IT management. Three improvements were deliberated (CMD 2 phase 2): a new revised design and implementation version of the MUST framework (A4), a practice of business plans (A5), and decision procedure improvements in the IT-steering committee (A6). To evaluate the implemented improvements (CMD 2 phase 3), participatory observations, workshops, semi-structured interviews, and meetings were used, which also link to the third CMD research cycle. As is shown in the timeline, the second action research cycle about organizational IT management overlaps with the third action research cycle about PD tools and techniques described below. To this end, the third action research cycle also supported evaluating of the workings of the organizational IT management deliberations.

In the third and final CMD research cycle, PD tools and techniques to improve how users on the shop floor could participate in infrastructure design were explored in the context of a comprehensive ERP project. This is accounted for in chapter 8 with the title "Let's talk about infrastructure." Taking on a PD approach about tools and techniques for ERP development was enabled by the results in the two previous CMD research cycles of deliberating a working approach to technical and organizational participatory infrastructure development. To understand the scope of existing practices going into the ERP project (CMD 3 phase 1), two additional shop floor development constituencies of the finance and human resource department were added to the ones accounted for above. The main objective of the action research became to deliberate (CMD 3 phase 2) the application of PD tools and techniques to understand users' present work practices, technical options, and new IT usage in an integrated technical and organizational infrastructure setting. With a particular focus on academic function in the ERP system, the action research supported design activities between the IT-steering committee, a number of shop floor development constituencies, and the IT-group. To evaluate the action research deliberations (CMD 3 phase 3), a number of retrospective interviews were carried out with key stakeholders. A second timeline was created with Evans (User) and Magnusson (User) and used both in a number of structured workshops and for informal feedback.

Attempts to abstract and generalize research results from the empirical research efforts were made to publish research results throughout the research process. Below, it is described how the chapters introduced in the timeline and described in the following chapters relate to these publications:

Chapter 5: The research setting and the organizational rationale of sustaining Participatory Design:

- *Version of chapter target towards PD and business models discussed in PDC 2012 workshop on business models, Johan Bolmsten*

Chapter 6: Shop floor users developing infrastructure:

- *Version of the chapter published in Bolmsten, J., and Dittrich, Y. (2011). Infrastructuring When You Don't - End-User Development and Organizational Infrastructure. In M. F. Costabile, Y. Dittrich, G. Fischer, and A. Piccinno (Eds.), (Vol. 6654, pp. 139–154). Presented at the IS-EUD 2011, Springer.*

Chapter 7: How Technology Matters

- *First version of the chapter presented at the 30th Information Systems Research Seminar in Scandinavia; revised version under review for the Scandinavian Journal of Information Systems*
- *Second version of the chapter submitted to Bolmsten, J., and Dittrich, Y. (n.d.). Technology Matters. Information Technology & People*

Chapter 9: Organizational IT management from the shop floor

- *Version of the chapter in review as a book chapter for EUSSET edited volume, editors Randall and Wulf.*

Table 4.5. Chapters published or in review

4.4. Summary and trustworthiness of the empirical research process

The embedded nature of the research gave access to people and organizational know-how that both helped to further the action research deliberations and support the triangulation and understanding of collected empirical data. This made the research pursued in this thesis potentially rewarding. As both Robson (2002) and Rönkkö (2005) point out, these are both important criteria in terms of bringing new insights and perspectives to the table that complement existing streams of research. However, incorporating a long-term embedded action research engagement that brings together both "use in design" and "design in use" dimensions in the research also posed challenges. Important design activities and interactions did, for example, not take place in a formally defined research project setting, which challenged empirical documentation. Even when projects, such as the course administration, were defined, they relied on a minimum of project documentation that could be used as a base for an empirical account and analysis. In addition, as Simonsen (2009) points out, compared to a case study, an action research project usually does not exist "out there." The researcher thus has to be engaged in both setting up and carrying out project work, making research both potentially time-consuming and risky. This also means that the researcher not only shares the ownership, but also has a stake in the project per se that is part of the action research deliberations. When working as embedded researcher, the challenges recognized by Simonsen (2009) can be argued to be even more present compared to a research design where the researcher can leave the research setting after a certain period of time for post reflection.

As described in the subsection above and summarized, care was taken to create a scientifically valid account of both the events and the actions taken in the organization, thus making it possible to argue for the trustworthiness of the empirical research process and data collection. To this end, the application of the CMD research cycle with its structured phases and guidelines was important.

In addition, as is shown in, for example, the timeline in Figure 4.9, the research allowed for gradually being able to iterate between different shop floor development constituencies and technical and organizational infrastructure entities. This was a fundamental component of building up an understanding of the empirical domain and deliberations. This can be related to Klein and Myers' (1999) basic principle of the Hermeneutic Circle that suggests that we come to understand a complex whole from preconceptions about the meanings of its parts and their interrelationships. Thus, the movement of understanding is constantly from the whole to the parts and back to the parts.

Below, the trustworthiness of the research is summarized along four dimensions; traceability, data triangulation, member checking, and researcher triangulation. As a complementary resource, Klein and Myers' (1999) principles for interpretative research are also referenced below.

4.4.1 Traceability

As complete as possible an account of the empirical research and its analysis are provided in the following chapters. To enable the reader to understand how the empirical research unfolded, chapters 6, 7, and 8 are structured in accordance with the phases of the CMD research cycle: understanding practice, deliberation of improvements, and implementing and observing improvements. A key component in this research is also that the different research contributions are grounded in a particular understanding of existing practices in their historical and situational context. These are based on opportunities and issues experienced from the perspective of shop floor development constituencies.

To make the chapters about the empirical research accessible to the reader, they are presented in a narrative ethnography format. This can be related to Klein and Myers' (1999) principle of contextualization described that critical reflection of the social and historical background of the research setting is required so that the intended audience can see how the situation under investigation emerged.

4.4.2 Data triangulation

The main documentation of the embedded research is through the research diary and project documentation. Where possible, the embedded research was complemented with participatory observations, workshops, interviews, and document and artifact analysis (data triangulation);

In addition, the use of the qualitative research tool, HyperResearch, provided valuable support for data triangulation of the collected empirical data. The analysis started with identifying codes in the collected material. Based on this open coding, a number of categories were developed that were used for axial coding, relating the different transcripts. Code maps were also developed to cluster different categories in relation to each other. In accordance to Klein and Myers' (1999) principle of abstraction and generalization, this facilitated relating idiographic details revealed by the data interpretation and analysis to theoretical general concepts in the discussion.

Additionally, it supported dialogical reasoning (Klein & Myers, 1999) to be sensitive to possible contradictions between the theoretical preconceptions guiding the research design and actual findings ("the story which the data tell") with subsequent cycles of revision.

Of further relevance to data triangulation and as part of the member checking, selected key members of the organization participating in the empirical research also participated in the timeline workshop, where empirical events and data were re-constructed and confirmed.

4.4.3 Member checking

The implemented changes as well as parts of the analysis were evaluated and discussed with the relevant members of the organization (member checking).

Given that the author of this thesis was an embedded research practitioner, for the trustworthiness of the research, reflection of the social construction of the data became very important. This translated, in particular, to inviting people in the shop floor development constituencies who were targeted to participate also in the methodological design of the research process. With reference to Klein and Myers' (1999) principle of interaction between the researchers, the CMD approach sets a focus on shop floor development practices. This sets the vantage point for critically reflecting on how the research materials are socially constructed through the interaction between the researchers and participants.

With each piece of research in this thesis, the number of shop floor constituencies, as well as technical and organizational infrastructures that were involved in the research, gradually expanded, both in terms of the number of people involved in a given shop floor development constituency and the number of shop floor development constituencies per se. This enabled complementary accounts of the same course of events. Already in the first action research contribution concerning how the technical base mattered, it was, for example, accounted not only for this researcher's perceptions as an IT-professional, but also for how a second IT-professional, Jönsson, worked with other local projects together with people working on the shop floor. We were both employed as IT-professionals with the same formal work description, but with different competency profiles that affected the work approach of how the development software support was carried out. Thereby, two complementary accounts of engagement with the technical base were included.

Getting multiple interpretations of existing practices was, for example, important to set the scene for action research deliberations. Klein and Myers' (1999) principle of multiple interpretations prompts sensitivity to possible differences in interpretations among the participants, as are typically expressed in multiple narratives or stories of the same sequence of events under study. This is similar to multiple witness accounts, who all tell it "as they saw it."

4.4.4 Researcher triangulation

Externally, supervising Professor Dittrich has continuously provided support for reflection on the social constructions of the research material through offsite debriefing sessions and implementing complementary interviews. Contextual specificities thus were elaborated throughout the research process. This can be referred to Klein and Myers (1999) principle of suspicion that requires sensitivity to possible "biases" and systematic "distortions" in the narratives collected from the participants.

As with all qualitative research, the results might not be easily transferable to other organizations. How the research contributes to related work is discussed in chapter 9. The following five chapters present the empirical research through the CMD approach.

5. The research setting and the organizational rationale of sustaining Participatory Design

The organizational rationale of engaging in action research to sustain a Participatory Design (PD) approach in the organization can be traced back to what kind of organization the World Maritime University is and the nature of the type of software development practices that hosted the empirical research. This chapter presents the characteristics of the research setting and how there was a need for staff to invent their own ways of working, including the development and management of software support. This premise for the action research of sustaining PD in the organization leads to taking advantage of the capabilities of end users to develop working software support. This usefulness criteria of both shop floor IT management and PD criteria is a redline that underpins the action research reported in the coming chapters. Applying PD is not only about enabling people on the shop floor to participate in design, but it is also connected to innovation and the creation of organizational value. This can be related to how Osterwalder and Pigneur (2010) relate the usefulness of PD to "the rationale of how an organization creates, delivers, and captures value." This can also be related to the distinction that Buur and Matthews (2008) make in terms of the difference between a PD focus on only design in the local project and "user-driven" innovation in the organization.

The account in this chapter is based on a series of ethnographical data collections conducted at the end of the research that enabled this researcher to step back from the embedded research situation and reflect on the broader context of the research setting. This chapter begins with a general description of the research setting that situates the way software development is carried out at WMU. The main empirical account in this chapter is from an interview with specialization professor Hughes about the evolution of IT capabilities at WMU and its connection to organizational IT management before the action research commenced. Hughes was one of the early employees at WMU that started to engage in software development to support not only course teaching in Maritime Education and Training, but also managing software development in the organizational arena. When the action research began, he had already retired from his fulltime position but still made annual visits as a visiting professor. The interviews with Hughes provided additional triangulation of the empirical data collection and evaluation of the action research.

5.1 The World Maritime University

WMU presents a research setting of a heterogeneous organization on several dimensions that has throughout its history been required to design its own mode of operandi both in regard to its educational offering and internal administration. This section describes some overall facts about the university's purpose and its operation that are intended to give the reader an opportunity to gain an appreciation of the overall research setting.

5.1.1 Some overall facts

The World Maritime University (WMU) is a postgraduate university. The university was established by the International Maritime Organization (IMO), U.N, in 1983 and operates under its auspices. Today, the university has an overall threefold focus on maritime education, research, and capacity building. The weighting between these three components has altered over time (and also depending on whom one asks). The university was, however, established with a strong focus on capacity building. The original purpose of the university is connected to the inherent international nature of the maritime sector, where trade has always moved around the different corners of the world. Making sure that all countries can manage their ports and ships, abide by the IMO legislative framework, and satisfactorily cope with safety and environmental standards are examples of global maritime concerns that make up the raison d'etre of the university. It is also a reason why building capacity in countries with a not already developed maritime sector is important.

To this end, a number of MSc programs are part of the principle educational offering. The programs offered are split between the main campus in Malmö, Sweden, and two "offshore" campuses in Dalian and Shanghai, China. In addition, the university conducts, for example, distance learning programs and professional development courses. A PhD program has also been added to the university's educational offering. Today, the MSc programs offered are Maritime Safety & Environmental Administration, Marine Environmental & Ocean Management, Shipping Management & Logistics, Maritime Education & Training, and Maritime Law & Policy, and Port Management (Malmö); International Transport & Logistics (Shanghai) and Safety and Environmental Management (Dalian).

In addition to the educational programs, an often-recognized dimension of the university is networking and providing students with relevant links to the maritime sector during their studies. The university, for example, has a comprehensive field study program, where the students visit different sites (shipping companies, ports, and agencies such as IMO) during their studies. These field studies are situated locally not only in Europe but also worldwide with travel destinations such as Japan, South Korea, Columbia, and Canada. The university has an annual intake of about 110 students in Malmö and 70 students in the China campuses. The majority of the students are maritime professionals when they come to the university, and just about all return to an immediate employment in their home countries when they graduate. The average age of the students is 34. In a typical year, students would come from 50-70 different countries; most of them situated in the developing world. In total, from 1983 to 2012, around 3,500 students from 164 countries and territories graduated from the university. Although the maritime industry has traditionally been male dominated, today, the university has a ratio between 2/3 male and 1/3 female students. A further opportunity for sharing maritime knowledge as well as cultural understanding and growth is enabled through most students living in the university's student residence. Today, the alumni community of WMU includes ministers, government secretary-generals, private industry chairmen, and managing directors.


Figure 5.1. Top-Left student group photo; top-right; students during a field study to a ship simulator in Germany; bottom-left graduating students

The university has a resident faculty and administrative staff of about 50 persons coming from 20-25 different countries. It is a strategic aim to recruit both faculty and administrative staff from different countries within the university's catchment area.

5.1.2 Organizational status

Although WMU is recognized as a specialized UN agency in Sweden, WMU is not directly part of the general UN framework – even if it is a subset of IMO – and does not directly abide by UN rules and regulations. As WMU is recognized as a specialized UN agency in Sweden, the university has the status of an embassy. This, for example, means that WMU operates outside Swedish law, tax, and social security systems. Furthermore, the university does not abide by any national or standardized university framework, which is otherwise mandatory in Sweden[1].

WMU is also not backed by any permanent financing body, but is responsible for securing its own funding. Principal financial supporters today are the government

[1] After the research was completed the university is undergoing an audit to become part of the European Credit system.

of Sweden (previously through its foreign aid program, now through the foreign ministry), the Nippon Foundation in Japan, and the city of Malmö. In addition, a broad range of government agencies and companies fund the university through, e.g., professor chairs and student scholarships.

Figure 5.2 Campus in Malmö Sweden

Figure 5.3. Resident professional staff members

These characteristics set WMU apart from not only other national universities in Sweden, but also from universities with an international profile. In relation to the latter case, a guest professor at WMU from Open University's MBA program made a number of reflections: even though the Open University also has a diverse staff and student composition, it was to her apparent how the UK way of doing things and the UK educational system denoted activities compared to what was done at WMU. The Open University, for example, had a special status amongst UK universities, and their MBA program is taught at remote locations in Europe and globally, with teachers having their origins in different countries with diverse educational and university systems. However, after having spent some time at WMU, she argued that the Open University came across as having substantial cultural similarities to the UK educational system, for example, of how to go about curriculum design and act as a teacher in the classroom. Similarly, there were set standards for the UK educational system of how to do grading and assessments, but at WMU there was a lack of such standards.

Especially the diverse staff composition of WMU and the special organizational status of the university, where there is a lack of set procedures that the university has to abide appear in the background throughout the empirical material.

People that have been part of the research put forward that these contextual characteristics are related to the shop floor IT management model and the pursuit of PD.

5.2 The rationale of the "native" shop floor development approach

From the empirical material, two examples are put forward below dealing with the lack of set procedures and processes and the staff diversity of the empirical domain contextualizing the empirical research carried out. These examples enable the reader to understand the research setting and the underlying rationale of the shop floor IT management approach

5.2.1 A need to invent procedures and processes

The following citations about how WMU had to design its own academic administration infrastructure from Registrar Davis is an example of the lack of set procedures and policies and how they had to be invented:

Davis: This is critical because, ahm, as I was going to show you [inaudible], when I showed up, we take a look at the graduates in 1996, that's what they got, after two years of studies, they got that, only thing it say is "very good", that was the grade report, after two years of studies, it gives you absolutely no information on what they did subject by subject

[…]

we had to come up with a whole system how you keep the grades, because you see, in the old days, I show you that, because we had a problem here, we didn't have any handbook when I came here [laughter], you have no idea, if I had any sense I would have left after six months to tell you the truth, because I could not believe it, and we had to fight so hard, you have no idea, fighting with the then president, with the staff, they didn't want a handbook

[…]

there was no credits, as you can see it was totally, 15, 25 [hours], I like 26 for that one, don't ask me why […] and it was an extraordinary battle, they hated, but then they really got in to it, they really liked it, the second piece was, we were [inaudible] and to be able to print this you have no idea, and then, we were moving along

[…]

this is what they are doing in the US, and I didn't push this, I was somebody, I was kind of reluctant thinking that they will not understand

[...]

but that was a titanic battle, and we were fighting another battle at the same time, there was about twelve different battles going on.

The above citations illustrate the context of where the shop floor IT management model originated, where Registrar Davis had to invent the very infrastructure of a socio-technical academic management system. The university was founded in 1983. By 1994 when Davis assumed his position, there was still not an accountable academic system. In looking through the background, empirical material collected, the reason for this state of affairs was not un-committed or incompetent faculty members or administrators – rather, arguably, the opposite. At other Swedish universities, this socio-technical invention of the complete registry infrastructure, (entailing not only the procedures of who was doing what, but the processes of what they were doing) would not have been needed or arguably even allowed. Generally, there are set standards and procedures of how an academic management system should work, with detailed requirements in relation to, for example, grading and enrollments. At WMU, the Registrar had to invent these bottom-up, together with the Registry Staff and other academic stakeholders in the organization. As further described in chapter 6, the development of the academic administration system became one of the first examples of shop floor IT management. Similar cases can be found in other domains at WMU that prompted shop floor IT management.

5.2.2 Implications of diversity of staff

The second example from the empirical background material relates to the first one, and highlights not only the lack of set procedures and processes but also the implications of staff diversity in their development, where faculty and administrators (as well as students) are from all corners of the world, bringing different academic traditions and best practices to curriculum design, teaching practices, and assessment standards.

When employed at WMU for the faculty to initially take on the faculty academic administration development, this researcher was aware of this matter on a superficial level. It was, however, not until a full immersion – a long time later – that it became clear that what was called an academic administration system at WMU was an institution taking on software support for teaching and learning.

Compared to industry standards, the academic administration development at that time was a hybrid between what is commonly a separate Learning Management System for the faculty and an academic administration system for the registry in other university settings. The reason for this mix was that Vice-President Wang (Academic) who was influential for the academic administration development came from a Chinese educational setting that was different from the US educational setting of Registrar Davis. For Wang, it was natural for faculty to have easy access to student performance measurements in order to be able to support the students to progress in their studies in a good way. For Registrar Davis this was off limits. In the US, it is a comparably important consideration that students' integrity should be protected from faculty and others. The frustration heightened for this researcher in his role as an IT-professional when he was asked to duplicate many of the student performance measurement functions in the registry system.

This example highlights not only the lack of set procedures and processes, but also how there were internal fundamental differences in opinions due to staff diversity about how they should be developed. Both these examples highlight the consequences of contextual challenges of the empirical domain on the research. On the account of people participating in the research, these contextual challenges related to the shop floor IT management model and also motivated a PD of technical and organizational infrastructure.

5.3 Organizational IT management before the initiation of the research

As has been made clear from the above description of the research setting, the development of software support - and later the action research - did not commence in a vacuum. There also were already established participatory decision making structures that worked as an organizational infrastructure for the shop floor IT management. The objective of the action research became not only to develop better integrated technical platforms (chapter 7), but also to develop better organizational IT management and representations (chapter 8 and chapter 9)

A further search into the historical computer development at WMU indicated that it "got off the ground slowly and steadily in 1983 in a piecemeal way for Wagner, the computer lecturer," under whose guidance a small computer committee was formed with users that had a stake and interest in developing software support for their work. This included both faculty and administrative staff. For example, during the start-up operations in 1983, a small number of computers and printers were purchased for word processing and administration needs. For the latter administrative purposes, administrative assistant White was a key stakeholder, who served in the computer committee (in chapter 6 it is further accounted for how White played a key role as a technical end user developer not only in appropriation, but in technical and organizational infrastructure development of electronic forms and databases as well as interfaced with people affiliated with the university such as staff and students).

There was also an interest from a number of faculty members to begin to use computers in teaching their subjects. A donation made it possible to set up the first computer lab and fit it with seven computers. And it was, for example, decided that student dissertations would henceforth need to be created by word processors (electronically inputted with assistance of secretaries). In the following years, the student laboratories were gradually extended and equipped. An additional donation in 1989 made it possible to purchase the first server and set up the first network. The network and server capabilities were then expanded at the beginning of the 1990's with first developments of email and internet links. This also called for the computer committee to develop the first staff and student computer rules and regulations.

Reflecting upon the growing importance of the computer systems to the operations of the university during the 1990's, the computer committee started to meet six times a year. In addition, the first student computer-working group was established and a student representative joined the committee. In the middle of the 1990's, a Network Specialist was also hired as the first IT-professional at WMU. In the interview, Hughes described the organizational role of people on the shop floor carrying software development and its connection to organizational IT management as:

Hughes: "I, I think, to be honest, you can say, that the computer committee drove the issues."

Hughes: "The trouble with management top-down is that you impose your own thoughts without having the grass roots views on what is needed, it is imposed on you" (computer committee members).

[…]

"it's also nice to have a guy at the top, but it might also be dangerous […], this is the way we are going to have it, and then force feed you, then you are in danger of not carrying your team up, so it needs to be interactive."

[…]

and you discourage this person from going the extra mile to develop things, knowing that, maybe this is not even going to get listened to; at least in the committee we listen to everything, all the ideas that would come in."

As part of a strategic review in 1995, a number of strategic recommendations for future action were made, including the standardization of software applications for all users and improved training for staff and students. To this end, White, for example, again played a key role in developing new procedures for Microsoft Office as a common technical platform and carry out training with both administrative and faculty staff (chapter 6).

As part of a second review undertaken by the computer committee in 1997, strategic recommendations were developed for computer installations and IT applications in classrooms, laboratories, for teaching methodologies, the use of the internet, intranet, email services, and library IT services. This resulted in classrooms being fitted with fixed computers and IT based delivery systems. In addition, to further the university's teaching offerings, a new multi-functional multimedia based laboratory with 20 computer interactively controlled from an instructor's console was developed. The new laboratory capabilities were, for example, used to educate students in distance-learning teaching methodologies in a Maritime Education and Training specialization. This enabled the students, who often were professionals at other maritime universities, to develop and try out the distance education approach with their fellow students. This was something that also became the topic of several student dissertations.

In addition, the university's English Study Skills Program (an entry program offered to all students), switched to a computer enhanced mode of teaching English. At the beginning of 2000, an additional laboratory was also added to support simulation based training, including a ship handling simulator and a number of workstations holding simulation programs covering cargo operations, terminal planning, voyage chartering, and communications.

The 1990s also saw additional software support developments on the administrative side, including an academic registry system and an integrated finance and salary system. As described in the above section, the registry system was also technically developed by the Registrar himself, as an end-user developer. The development was based on a high-level programming-language and database called Visual DataFlex. The registry academic administration system was developed to meet requirements of both the registry and faculty of record keeping and institutional reporting needs in relation admissions, student affairs, grade recording, graduate directory information and donor and external contact address tracking (chapter 6). In addition, an early version of the academic administration system for the faculty had been developed by Wang in his earlier position capacity as a specialization professor. This development was later overtaken by the IT-professional, Nilsson (chapter 7).

What this expose shows is how the development of computer and software capabilities was organizationally managed by faculty and administrative staff themselves. This was the setting in which the action research of pursuing the management of sustainable PD took its stance. All the following chapters (7, 8, 9) that account for particular episodes of action research begin with a vignette that shows how the research begins with understanding practices and problems from the practitioners' point of view.

5.4 Summary

This chapter has described the research setting, the rationale of the "native" shop floor IT management approach, and the original organizational IT management structures.

The research of this thesis took its stance in these grounded realities when this researcher was employed as an IT-professional at WMU. Pursuing sustained PD was thereby not a construction invented for the purposes of research alone, but about working with the reality of the organization. In this way, this chapter has set the scene of the usefulness rationale of PD, that is, of going about the development of software support at WMU. This runs as a redline through the action research guided by the CMD methodological approach, with emphasis placed on being continuously grounded in organizational realities of shop floor people. How the action research supported the development of organizational IT management was, for example, based increasing needs for integrated technical infrastructure (chapter 7), and there was a need to improve the procedures and processes of the computer committee as an organizational infrastructure to the shop floor development constituencies (chapter 9).

6. Shop floor users developing infrastructure

This chapter concerns the original shop floor IT management approach at WMU, and how it is connected to infrastructure development. This shop floor IT management approach is traditionally based on End-User Development (EUD). At WMU, end-users take on an expanding role as technical developers. This relates to the bottom-up tradition described in the previous chapter about how software support has traditionally been managed in the organization. In this way, EUD allows users on the shop floor together with their colleagues to develop and evolve their own IT-tools to support work tasks to become more effective in their day-to-day work. EUD furthermore allows innovations with respect to processes and work practices to be mapped easily into the supportive technology.

At WMU, shop floor IT management is an approach to software development that is encouraged. The account of the original shop floor IT management that is based on EUD is complemented in the forthcoming chapter with shop floor IT management that also involves IT-professionals. As described in coming chapters regarding integrating different dimensions of technical and organizational infrastructure, although there are practical challenges of maintaining a shop floor IT management, the opportunities are still believed to outweigh them.

The account given in this chapter is based on an ethnographic study of shop floor IT management of a registry, administration, and electronic address database infrastructure. Given the order of the empirical research, it was clear when initiating the ethnographic study that people in shop floor development constituencies at WMU were influential in participating in the development of their own software support. However, what came as a surprise in the two cases described here was the extent to which people on the shop floor were not only appropriating and developing specific local application, but, in fact, were also taking on the development of technical and organizational infrastructure themselves – with and without the support of IT-professionals. Before this ethnographic study took place, this researcher had mainly worked as an IT-professional for the faculty with academic administration support. In this way, the research carried out here also became a way to triangulate the action research accounted for in the other chapters.

In the context of the other research contributions of this thesis, the empirical results in this chapter thereby contribute to an understanding of the role and value of

EUD when moving towards a more integrated technical and organizational infrastructure to consolidate the separate systems. EUD has been part of the organizational ICT development practice from the very beginning. The cases that provide the empirical basis for this chapter have been selected, keeping in mind that the End-User developers have managed their software development on the shop floor for more than 20 years. Their development activities have been acknowledged as being important for the organization. As the scope, technical sophistication, size, and character of the user community differ significantly, the cases together provide a consolidated picture of shop floor IT management based on EUD at WMU.

The remainder of this chapter is structured as follows: Section 6.1 presents the two empirical cases. In the following section 6.2, the analysis is presented, where the challenges of EUD in infrastructure settings are developed with respect to five aspects: organizational support for EUD, cooperation with (other) users, cooperation with professional developers, coordination of EUD and professional IT development, and technical platform. The conclusion summarizes the contributions made.

6.1 The Two Cases

6.1.1 Case 1: Electronic Forms and contact database

The end-user developer of the first case is White, a long-term and current senior administrative assistant at WMU. White has been a member of staff for almost thirty years and has been part of the university's journey from a manual typewriter operation to an increasingly integrated technical infrastructure. Talking with her, one recognizes her genuine interest in smart solutions, which save time and effort, as is evident when she refers to her first encounter with computer based forms: "So I learned that you can do online forms […]. I thought this was just the best thing since sliced bread."

Her role in developing IT support for administrative purposes for the whole organization is acknowledged, but not organizationally defined in, for example, her work description. Referring to this semiformal position, White describes herself as "sort of a spider in the net." For eight years she was also a member of the computer committee that gathered key shop floor development users, IT-professionals, and managers deciding on the IT infrastructure..

Two of White's areas of responsibility are to administrate: (1) internal forms such as leave and travel requests and (2) a repository of WMU contacts – initially only for administrative purposes and later across all university functions. From the beginning, these were based on paper and typewriter. This started to change when word processing programs with contemporary features became available. White especially recalls version eight of Word Perfect where it became possible to set up electronic forms:

"We are going back 20 years you know. [...] I thought it was super [...] So I use help a lot, and I have learned to read the screen [...] I went through it step by step you know. Click on the name, textbox and fields, and all that you know. I learned about the fields. Trial and error, first it didn't work you know. So I made a leave request form [...] I take a form that is for everybody, then you get something that is across the board. And my boss at that time [...] I tried it on him of-course."

White ended up not only migrating the leave request form to Word Perfect but also the rest of the administrative forms.

In addition to being used for electronic forms, contacts also started to be maintained in Word Perfect. The problem was that many experienced this approach to be insufficient. Contact information became scattered throughout the organization, and in order to get hold of information about a certain person, one had to know who internally was maintaining a particular record. A discussion started about the benefits of having a central and standardized point of reference for contacts: a database with generic and standardized fields appropriate for different functions that anybody could access and query. In the end, it was decided to go ahead with the implementation of a Microsoft Access database. To internally be able to develop and maintain the database, all the administrative assistant and secretaries were sent on a Microsoft Access short course. Upon their training, White - then already known for her technical interest and expertise - ended up taking charge of the development of the contact database, creating both the database and associated interfaces. The idea was that the secretaries would primarily be in charge of inputting data, whereas professors and others also could extract it. In the end, the contact database contained altogether about contact 640 records.

For both the electronic forms and the contact database, White gradually developed a model for user involvement. In regard to the electronic forms, White describes how she worked actively getting feedback from other users. Acknowledging a wide range of competencies, she developed an implicit ranking of users from computer illiterate to technical experts on which she tested prototypes. Already when migrating the forms to Word Perfect, White started to work with different colors, fonts, and layouts to make the user experience more intuitive for the different user groups and purposes. In regard to the contact database, she produced manuals and trained the other secretaries in using the interfaces of the database.

Both the electronic forms and the contact database have undergone major revisions. For the electronic forms, the next major technical infrastructure change was an organization wide change to the Microsoft Office suite and Word. For White, this meant that it was back to the books and the help files to learn. The Word version of the forms had been in operation for 18 years and became the de facto standard in the organization and also part of other technical infrastructures such as the web-based intranet. However, although new and advanced technical features became available with the Word-based forms, such as mail merge and calculation capabilities, White was never altogether satisfied with the format. She experienced Microsoft as more "fuzzy […] if you are a new user to forms." The latest revision embarked on involves using pdf and adobe life cycle as a technical base. This change enabled full integration of the forms with other applications, allowing White to continue to improve usability aspects. For the technical integration, White had to learn how the XML based backend of the forms worked. She has since then been involved in creating several prototypes in cooperation with one of the IT-program officers, where the forms exchange information via web-services with the in-house intranet. In addition, the easier design of the pdf forms has opened up for White the possibility of training other end-user developers to create their own forms.

Comparably, the biggest infrastructure changes to the contact database have been of an organizational nature. Coming up to four years in operation, the contact database and White's role in developing and maintaining it became subject to fluctuation. One after another, the other administrators retired or left the organization. At the same time, the university started to employ IT-professionals. This, for example, meant that White for the time was relieved from the coordinating computer committee on behalf of the dedicated hired IT professionals, such as the IT-Program

Officer (faculty). Gradually, this implied a disruption in the organizational anchoring of the contact database in regard to use and development. This did not mean that the Access database lost its organizational relevance or that White stopped developing it, but it was turned into a dedicated address database with White herself as the main user. During the last years, a discussion has again emerged about the benefits of having a central contact database. The first pursuit to re-establish such a database, has come with the development of a new external website. Using the contact database as a foundation, White has again become involved in implementing new features and updating the contact records. The intention is that these records subsequently will go into an ERP system in the pipeline for implementation.

6.1.2 Case 2: The registry system

The registry system for WMU was developed from scratch by Registrar Davis. Davis came to WMU from the United States in 1992 and then already had comprehensive experience of the function from American universities. In the United States, an often mandatory requirement of the Registrar is to have technical expertise. Being able to operate databases with connecting reporting tools in order to, for example, provide a decision support of student data is a fundamental task. Today, the most common off-the-shelf system is Banner. When Davis initiated his career, what today are common features in such systems were then still in a pilot stage. In fact, before Davis started his employment at WMU he participated as a domain expert in the development of an early registry software tool. During this project, he also managed to advance his technical expertise by gaining his first experience of high-level programming.

The registry function at WMU is modeled partly after the US system, and the Registrar holds a managerial position on the same level as a Vice-President. The registry department at WMU is made up of four employees: the Registrar Davis; Associate Registrar Jackson; Student Services Officer Evans; and the Senior Registry Assistant Magnusson.

In 1992, WMU did not have satisfactory university standards for core registry functions such as course, subject, credit, and grade management. Instead of subjects that had a direct relation to weighted credits, courses were made up of modules that defined broad teaching areas. The modules were not individually graded, and the certificates presented to the graduates contained only an overall evaluative statement "Can you imagine, coming into this situation?" Davis reflects.

When Davis initiated his employment at WMU he gradually started to construct an accountable academic management system. Alongside this, he also began to design and implement a computerized system himself. Building on his previous experience, he picked a high-level programming language and database called DataFlex together with the reporting tool Crystal Reports. After a number of generations of the system – the grade management has undergone eight successive evolutions – Davis believes that he has succeeded towards accomplishing his initial vision: "What we do is we built the system to basically do our jobs, all of our jobs, all the four people in the registry, and that was exactly the purpose, to go from a manual paper based operation, to a computerized electronic method." Today, a dedicated and tailored computer support is in place for major WMU registry functions such as admissions, student profiles, courses and subjects, grade management, and quality assurance. The system has come to contain additional components such as alumni records and hostel management.

Despite the vision of a comprehensive computerized system, some processes still need a combination of electronic and hard copy operations. The start of the admission process is, for example, marked with the registry department receiving a paper application form. The data are transferred into the registry system. Thereafter, the application is subject to a complex admission process involving both internal committees and external agencies, which is supported and documented in the system. Once the student is admitted, all study activities and results are documented as well.

Magnusson has been one of the main users of the system cooperating with Davis around the design of it. She recalls the evolution of the data entry interface: "I know that in the beginning, when I started, these tabs where divided in three different databases, and I thought it was rather complicated to remember which tabs that belonged to which […] you can always call him, go in to him, and he listens […] it is not like it is a small petites, he does do, writes it down on his little notepad. I have not thought about it before, but now when we are talking about it, it is pretty great […] and then he either says it works, if it works […] when he says it doesn't, it is because it must be possible to extract some report."

Members of staff outside the registry department, though, were less satisfied with their access to the registry system. They could not, for example, extract reports from the registry system, apart from a number of pre-defined template reports, without acquiring substantial knowledge about the database structure and the report tool. According to Davis, he down prioritized requirements from outside the registry department due to time constraints: “The whole concept behind this wasn’t to be for the university, it was supposed to be for the registry only, and then we decided to give it to people, it wasn’t meant to be the ERP system for the whole university, it was for us to get our work done, and then people wanted things so, I then, I had to go in there, and then they were never happy, the main thing is I would have had a full time job just coding this.”

6.2 The role and value of End-User Development in infrastructuring

In this section, both the field material and the findings of the related analysis are presented. The subsection headings were derived from the field material in the manner described in the research approach chapter. Each subsection starts with introducing the theme, after which an account of the findings from the empirical research is provided and evaluated.

6.2.1 Maneuvering as an informal developer

As an informal or semi-formal developer, end-user developers can be in a vulnerable position. On the one hand, they develop part of the IT infrastructure for the whole organization and provide important tools. They are aware of their role, and, e.g., consciously include relevant stakeholders. On the other hand, as the episode with contact database shows, they are not officially recognized as developers, and other organizational actors might not be aware of their activity, especially when personnel changes.

White: an unprofessionally professional developer. Even though White is one of the main beneficiaries of her work, she deliberately targets other staff with her development. She not only gathers ‘requirements’ in an informal way but consciously addresses lifecycle management such as training, further development, and maintenance. She, for example, not only develops the electronic forms, but runs informal user tests and provides help. The contact database comes with a user manual. Both are maintained and adjusted to changing requirements and technologies. However, although, the nature of her IT-development in many ways is closer to that of an IT-professional than a ‘normal’ end-user developer, she is only informally recognized as a champion user in the organization. IT-development cannot be found in her work description. Acquiring new IT competences is often done in her spare time and she, for example, carries expenses for books. Although her efforts are appreciated, her ability to maneuver in the organization for good and for bad is affected by her “unprofessional” status. Whereas her IT-professional colleagues are permanent members in IT- forums, such as the computer committee, this is not the case for White.

The necessary cross-departmental IT-coordination takes place in a different way: White describes herself as “sort of a spider in the net” when it comes to development and coordination of the IT related ventures she has been involved in. This allows her to continue being part of the forms development and the contact database, but that the coordination has a different shape outside the formally arranged forums. In the latest attempt to revive the contact database in the context of a new external website, it was WMU’s president that turned to White to assist with the coordination. The reason is that through her day-to-day work, she has an established relationship to internal staff stakeholders and knows whom to ask for requirements and how different people could contribute. White has continued to maintain her relationships with the professional IT developers and thus acts as a broker between the users and the IT developers.

Davis: “The captain that controls all the pieces” Also Davis is not explicitly recognized as an IT professional at WMU. However, as a Registrar, he is a senior management member. With respect to his development mandate, this implies that Davis has space to basically carry out development for the registry system in the way he sees fit, as long as his department meets the university’s overall expectations. To this end, Davis has also taken on tasks beyond the ‘normal’ end-user developer enhancing of individual tools. Even though he is a central beneficiary himself – throughout the interviews, both Davis and Registry Assistant Magnusson continuously emphasized that the core function the of the registry system is to be able to output the right type of student management reports – the client interfaces and integration with other software, such as mail merge for Microsoft Office, are generally more used by the other registry staff.

In his capacity as a management and (recognized) key domain expert of his department, Davis has a permanent place in the computer committee. However, except for securing his annual development budget, his use from the committee is limited. Davis' vision is to build an "electronic method" for all core functions of the registry department, but only for the staff members of the registry department. Today, the registry department has the most comprehensive support. At the same time, all input into, and export of, information beyond the department is done manually, e.g., grades arrive to the registry department in an electronic format, but have to be manually transferred one by one. Even though some client interfaces for faculty exist, de facto, people call Davis and ask for different reports to be exported.

During the interviews, the possibility of integrating the registry system with the surrounding infrastructure was discussed. It turned out that this would have been technically possible. However, the protective attitude of Davis – which enabled him to develop a comprehensive and consistent application – hindered an earlier exploration of such possibilities.

End-user developers can have a vulnerable position in an organization if their expertise regarding both their development tasks and the organizational needs are not recognized. When establishing an organization to coordinate infrastructure development their 'shop floor IT-management' and their ability to act as brokers between users and IT professionals need to be understood and recognized for the benefit of the organization. Adequate forms of representation need to be established.

6.2.2 Frontline user cooperation

When end-user developers develop systems for others to use, cooperation with these other users is important as well. Not surprisingly, both the end-user developers interviewed and observed have an established practice of involving other users in the development.

White: An End-User Developer learning about usage. Rightfully, White describes herself as "one of them" – her users – and claims that she has a good conception of how the contact database and the electronic forms will be used. However, instead of only using herself as a reference user, she also works actively to understand the perspectives of other users through for example prototyping and testing against different stakeholder segments. The reason is that she is directly confronted with the problems other users have with her applications. As an administrative assistant, she is placed in the middle of organizational activities with long-term established relationships with other staff. She is, therefore, one of the first to get notified if her development ventures do not work: “I end up with more questions then, and if there is more questions I end up with people who don’t use it.” And people “who don’t use it” mean more work for herself.

Davis: Caring for his users. The motivation for Davis to involve his department in the development is a different one. The change, e.g., initiated by Magnusson was not a malfunction per se. The program was fully functional. Its prior design was developed in accordance with the preferences of the previous registry assistant. The changes that Magnusson called for involved Davis changing both the interfaces and the database.

During the interview with Magnusson, she compared the way Davis cooperates with the members of the registry department to previous experiences. She worked as a secretary at a major company during a migration to SAP: “I mean, there was never any question of us having any input to it. It was like it was, but they had some sort of groups, from different departments where they went through what was needed. But then afterwards, it was like it was. […] But I guess, there are pros and cons with everything.” One con is raised against Davis’s way of development: “Honestly speaking, it can appear a bit stiff, for example, you have to save here, there. One perceives it as a bit old fashioned when one enters information.” Given the limited development resources and the resulting need to prioritize, this does not come as a surprise.

End-user developers care about usability and they are confronted with the problems of not usable software. Their expertise can be used by professional developers when working with IT infrastructures: As members of the user community and as shop floor IT managers, they might be able to help with recruiting the right people for user participation and also be able to prioritize between crucial problems leading to users refusing an application and ‘good to have’ features that can wait until developers have time.

6.2.3 From one software developer to another

Modern IT infrastructures for educational organizations with needs to support both external and internal cooperation are not possible to maintain without professional IT developers. WMU has decided to have IT competences close to faculty and administration. Over the last seven years, two fulltime positions have been established. This requires the end-user developers to cooperate with their IT professional colleagues.

White: Including the professional developers in her network. IT professionals are colleagues too. White's way of managing the professional IT developers is to include them in her network. In this way, she is consulted and included in the development interfacing with, and impacting, her applications. In the case of the pdf forms, she negotiated the backend development with the IT-professionals in order for the results to be compatible with a future integration in a wider IT-infrastructure. In regard to the updated contact database, she had to coordinate the interface development with both professional developers and contributing staff members. As the integration into the infrastructure poses new technical problems, this cooperation includes opportunities for learning new technologies.

Davis: Isolating the own application. Due to the need to limit the complexity of an already complex system, Davis isolated the own application from the development of IT-infrastructures around the registry department. One result of this is that the possibility of technical integration has not been explored.

In the context of infrastructure development, the cooperation between professional IT developers and end-user developers as well as users is important in order to coordinate more substantial development by the professional developers with the EUD parts of the infrastructure, i.e, evolution of the electronic forms and the contact database needs to be coordinated with the Infrastructure development. The formal organization of the IT infrastructure development needs to accommodate the need for coordination and cooperation between professionals and EUD.

6.2.4 … something that otherwise would be defined as a project.

The need to coordinate EUD and professional development of the same infrastructure has been highlighted above. However, as is already indicated in previous research (see, for example, the gardening metaphor in Henderson & Kyng, 1992, and also Kanstrup, 2005), EUD often takes place without a formal project organization, and is interlaced with the actual tasks of the end-user developer.

White: Focusing on specific applications. Both the development of the electronic forms and contact database would normally be defined as projects, except that in White's case, they do not qualify as such. At least not according to what can be recognized as traditional IT-project criteria, such as predefined scope, resources, and start and end point. The scope is negotiated between White and her users; resources are found whenever there is no other urgent task, and the whole ends when there is no one using the results any longer. There is no project charter, no formally defined objective, nor identified constraints and stakeholders. Even the implementation platform changes over time. However, both development activities are clearly limited. The forms development is about administrative forms. Requests to develop forms for other departments are answered by teaching the person to do it by him or herself. The contact database is about people and addresses. Other functionalities vary over time.

Davis: developing for the registry department. Davis did not organize even major revisions of the registry system in any formal way. When Magnusson was asked about how improvement proposals were handled, she answered: "Eh, I don't know, I was just happy that it was reduced [in reference to the databases connecting to the tabs and fields]." The developments of the registry system and the cooperation between the four staff are not done with a formal project management or a project charter. Judging from the interviews, the question remains whether such measures would only have been bureaucratic red tape. The development seems to be coordinated by informal meetings. However, Davis clearly limits his development activities to the support for the registry department.

As EUD does not occur in the form of projects, professional development needs to develop ways to coordinate infrastructure development with the more flexible ways in which EUD takes place. Formal committees, such as the computer committee, provide a place where some of the coordination can take place. However, it is not certain whether simply providing a meeting place will be scaled when the organization grows beyond a size where professional and end-user developers can sit around one table.

6.2.5 Technical platform

From the end-user developer's perspectives, the technical platform provides challenges; from an organizational IT infrastructure perspective, it both enables and constrains development. Below, the relevant parts of the field material are summarized and the implications are discussed again.

White: combining reading manuals and trial and error. How do end-user developers acquire the necessary competences for developing stable applications for usage by others? White applies two strategies: Continuous trial and error and step-by-step development is used to solve technical problems arising from her everyday work. In the end, this leads to the intimate knowledge of the workings of a technical platform: "I know how they were thinking when they made it [MS Word]." The other strategy is to acquire more abstract knowledge. White herself emphasizes the need for her own diligence. When it comes to reading books and manuals to learn about the existence and properties of technical features and possibilities before and during her development, she reads up on the workings of a new feature, makes a small prototype for testing at home, and then transforms that into a working functionality in relation to a current task.

White's ability to assimilate de-contextualized, technical knowledge and put it into practice allowed her to port the electronic forms across three different technical platforms: from Word Perfect to MS Word and to the adobe suite. The last change provided an additional dimension. As the forms now should interface to databases and other applications, the data model behind the forms needed to be more independent from the form seen by the user. To cope with this challenge, it was necessary to understand notions such as data structure and mark-up languages (XML). The cooperation with one of the professional developers helped to master this learning step.

Davis: Technical proficiency as part of the job description. That Davis is a technical domain expert is not so strange: Technical proficiency is “the first thing they list in any job advertisement” in the United States. This might not necessarily include programming expertise. As a small institution, it would not have been feasible for WMU to purchase a comprehensive standardized system that then had to be customized to fit the very specific needs of this university. Practically, the majority of this code is developed by Davis himself. In addition, DataFlex has an active community that contributes with script that has been incorporated to create more advanced menu structures.

The implementation platform, respectively, exchanging it, shows in the empirical material as technical challenges. Interfacing EUD results to an infrastructure does contribute additionally to the requirements for technical and conceptual know-how. From an infrastructuring point of view, the implementation platform for the infrastructure has to be selected carefully to provide the possibility to interface to heterogeneous applications and to allow for non IT professionals to use it for a base for EUD. (See the following chapter for a further discussion.) The interfaces between EUD results and the infrastructure indicate where coordination between professional and end-user developers is needed. When evolving and introducing new technical platforms, however, impacts on the EUD results need to be considered and the end-user developers need to be provided the necessary support to update their technical proficiency.

6.3 Summary

This chapter has addressed shop floor IT management, where the relationship between EUD and organizational infrastructure development and evolution at WMU has been in focus. Based on the evaluation of the empirical material, it is concluded that EUD and organizational infrastructure development can be combined in shop floor IT management. As also described in the previous chapter, this can carry important opportunities to develop useful software support. The meeting of these two different practices of development is, however, at the same time not necessarily an easy one, and poses challenges to the deliberations described in the coming chapters regarding integrated technical and organizational infrastructure. In the evaluation of the empirical material, dimensions were outlined to consider that: the role of end-user developers and their representation in the strategic committees need to be decided; and end-user developers' expertise in shop floor IT management and their established role as brokers between professional developers and users can provide a resource for professional infrastructure development as well.

Cooperation with professional IT developers, however, needs to be fostered. The interface between project based professional development and EUD is an open challenge, where joint participation in strategic committees can be seen as a starting point to explore ways to coordinate. This is the focus of the second initiated Cooperative Method Development action research cycle reported in chapter 9. Finally, the technical platform connecting the heterogeneous applications provides a challenge for many end-user developers, who might need help to conquer this new technology. The platform needs to be selected so that independent applications can easily be included and that end-user developers can use the platform for their own development. This is reported in the following chapter about the first Cooperative Method Development action research cycle of how the technical base matters in infrastructure development.

7. How technology matters - The role of the technical base in infrastructure development and evolution.

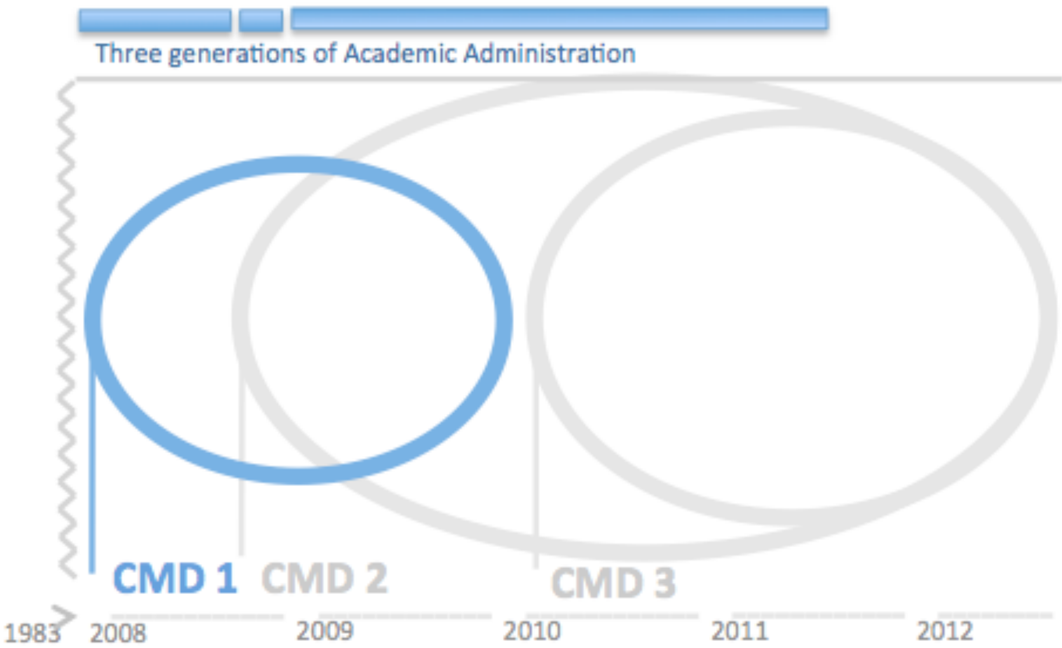


Figure 7.1 The first Cooperative Method Development Cycle action research cycle indicating how a shared technical base requires participatory organizational IT management coordination. (see the complete timeline in chapter 4)

Whereas the previous chapter described how shop floor IT management is an important capability, also connecting it to infrastructure matters, this chapter relates to how a technical base links development of software support in different individual shop floor development constituencies. Here, the technical infrastructure is in focus and how it enables and constrains the space for sustained Participatory Design (PD) in relation to different shop floor development constituencies. The fact that different technologies have different affordances as a technical base should come as no surprise. How different technologies implicate not only design processes and outcomes in a socio-technical evolution of an IT-based infrastructure but also for PD are, however, under-researched themes (Orlikowski, 2010; Pipek & Wulf, 2009). In the action research deliberations reported in this chapter, better advice to this end would have been helpful.

7. How Technology Matters - The Role of the Technical Base in Infrastructure Development and Evolution.

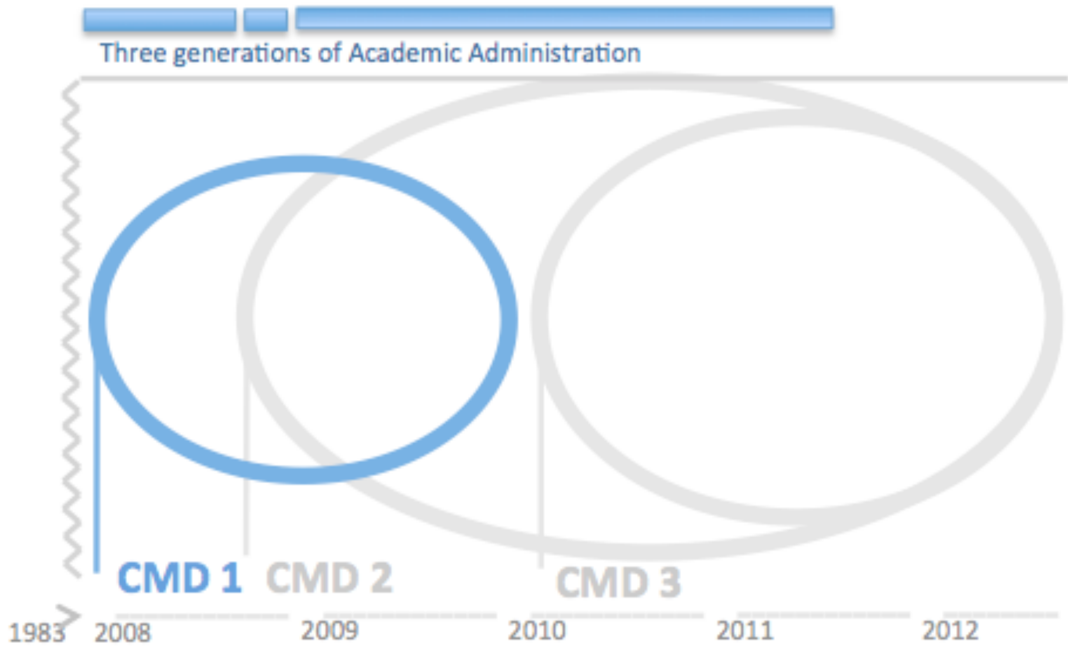


Figure 7.1 The first Cooperative Method Development Cycle action research cycle indicating how a shared technical base requires participatory organizational IT management coordination. (see the complete timeline in chapter 4)

Whereas the previous chapter described how shop floor IT management is an important capability, also connecting it to infrastructure matters, this chapter relates to how a technical base links development of software support in different individual shop floor development constituencies. Here, the technical infrastructure is in focus and how it enables and constrains the space for sustained Participatory Design (PD) in relation to different shop floor development constituencies. The fact that different technologies have different affordances as a technical base should come as no surprise. How different technologies implicate not only design processes and outcomes in a socio-technical evolution of an IT-based infrastructure but also for PD are, however, under-researched themes (Orlikowski, 2010; Pipek & Wulf, 2009). In the action research deliberations reported in this chapter, better advice to this end would have been helpful.

The first Cooperative Method Development (CMD) action research cycle is reported, that is, how software support for academic administration based on Web 2.0 technologies becomes part of the technical and organizational infrastructure, and as a technical base, enables and constrains how IT-professionals can work with shop floor users in the development of useful and usable software support. As at universities and organizations in general, information technology is being used more and more as an infrastructure where different applications and modules are integrated in a flexible way to provide support for groups of users cooperating across the whole organization.

The development of a web-based support for the planning, administration, organization, and implementation of courses is of importance at WMU. Faculty, study administration, examinations office, and even students need to cooperate. Different functionalities supporting the tasks of the different stakeholders need to be integrated but also have to be evolved due to changes in study regulations, the organization, didactical improvements, and also changes in work practices. What is needed is a technical base allowing the flexible integration of heterogeneous applications and modules supporting this evolution. The focus of the action research in regard to the development of academic administration software support from the beginning has been to adapt PD to support infrastructure development.

This chapter describes shop floor IT management at the faculty of WMU, which is the base of the action research deliberations. The organization of this shop floor IT management complements the type of shop floor IT management described in the previous chapter, in that IT-professionals are added to the community of users and end-user developers. The faculty shop floor IT management is the primary anchoring of this researcher's work as an IT-professional at WMU. The following section sets the scene for the action research by introducing the development practices and evolving needs of the faculty shop floor development constituency in regards to academic administration software support. Three evolutions of the technical base are then accounted for as part of the action research. This is followed by a combined analysis of the changes made where key considerations of the technical base in relation to PD are highlighted.

Web 2.0 infrastructure

The technical base that is subject to deliberations in this chapter uses Web 2.0

technologies. Web 2.0 describes websites that are based on dynamic functionality, where content can change in the interaction with the user. This extends the opportunities of website development from earlier static websites. There are many technical bases using Web 2.0 technologies, all of which put forward their distinctive affordances. A overview of these can be found at this link:

http://en.wikipedia.org/wiki/List_of_content_management_systems

In the action research reported here, a number of Web.2.0 technical bases using Microsoft ASP and ASP.NET programming languages and architectures are used However, in regard to Web 2.0 technologies and contrary to the argumentation here, it even appears that there are emerging contributions arguing that technical base considerations become even less important in user-driven design, given the new qualities of such technologies. For both action and research purposes, this fact makes it even more relevant to pose the question of this research study: can we, for example, hope that the technical qualities of Web 2.0 are "seemlessly" flexible - almost a generic quality to the benefit of PD and End-User Development (EUD)?

Floyd, Jones, Rathi, and Twidale (2007) argue that the capabilities becoming available with Web 2.0 technologies are driving the development of new software development practices and have the potential to change our mindset on how to do application development. Only through, for example, the possibilities of increasingly basing development on the integration of already existing data, services, and components such as web mash-ups ideas, can we rapidly realize what would otherwise be too time consuming and expensive to pursue. According to Floyd et al. (2007), using the capabilities of web mash-ups facilitates PD techniques, such as patchwork prototyping that has been the holy grail in prototyping research over a long time. With web mash-ups, it is possible to create and rapidly iterate high-fidelity prototypes that make it easier to bridge the traditional division between design and use, as it is possible to immediately incorporate working prototypes in the end users' daily work activities. As prototypes can be easily switched turned on or off, or be reconfigured, it is argued that we can rapidly evolve them into useful working technical artifacts in action.

7.1 CMD phase 1: A need to extend the "native" shop floor IT management to Participatory Design

This section deals with the first phase of the CMD action research cycle and sets out to give a basic account of the academic administration development practices in their historical and situational context, and to identify fundamental issues as a base of change.

The start of the development of the academic administration system marked the start of an additional shop floor development constituency at WMU with its own shop floor IT management approach. As described in the previous chapter, both the development of the registry system (managed by the Registrar Davis and the registry staff) and the development of the address database and electronic forms (managed by the administrative assistant White) show other instances of shop floor IT management.

Vignette 1 describes how the development of the academic administration system was initiated, which also comes to constitute a platform for how the IT-professionals at WMU started to engage in technical and organizational infrastructuring activities beyond the local project.

<u>Vignette 1: From arbitrary PD to PD 'one to one'</u>

"When I joined WMU, I start to use my excel thing to help me to manage my specialization […] At that time [it] fascinated me, you can link in computer […] If you see, in one sense, excel is a database […] you play with excel, then this notion establish in your mind".

This was how Wang started to develop software support to manage his courses as a specialization professor in 1998. He later went on to "play" with HTML and Front-page and even bought the books for ASP. Although his research showed "I have to use this ASP", he also realized "I am not the person to it".

When Wang was appointed Academic Dean in 2001 an opportunity arose to hire the IT-professional Nilsson to from the start work dedicatedly with developing dynamic Web 2.0 ICT academic administration support. Wang describes how he used his experiences with excel as a foundation to explain what he wanted and how a close collaboration was initiated: "We had almost daily conversations on how to develop this and that…so for example, I wanted, this online marking, distribution of marks, *but he came up with a product which goes beyond my expectation, he then worked out to give the individual marks to the students individually, this I didn't think, but he made it, he showed it is possible, because he saw our three secretaries, every week busy to type one small piece of paper, This I didn't think, I could figure out that was possible, but he made it. He showed it is possible"*

The people of the academic administration shop floor development constituency did not know the term PD, nor had they encountered any PD methods, tools or techniques. During the analysis below, the somewhat ambiguous notion of "PD one to one" is used to describe this unwitting PD practice: Wang, in general, wanted to promote an approach where software support was developed in-house and in close concert with users' work realities. This was also visible in the placement of the IT-professionals at WMU: Nilsson, for example, shared an office with two of the faculty secretaries with also partly overlapping work areas, where he was conducting administrative work in addition to IT development. Moreover, as Wang stated above, he believed that part of Nilsson's success with the online marking was to be attributed to Nilsson, working closely not only with him, but also with the faculty secretaries.

In other words, there was an ambition of a PD prone development style. The "one-to-one" addition is called for, as there was no systematic organizational inclusion of different stakeholders in the process. When this researcher was employed in 2005, the need for a more systematic approach to participatory development of technical infrastructure quickly became visible. When starting not only developing individual applications but integrating them into an infrastructure supporting collaboration across different roles and department, the newly instantiated IT-group faced the challenge to both technically and organizationally support the shop floor IT management of different shop floor development constituencies. This development then became subject of the action research study that is reported here.

7.2 CMD 1 phase 2: Participatory Design and the evolution of the technical base

This section deals with the second phase of the CMD research cycle. Being the first action research engagement of the PhD research, the deliberations of improvements were comparably the most open ended. It turned out that there was a range of issues in regard to PD and where the technical base was of relevance both to the organization and to the research. As indicated in Figure 7.2, in total, three improvements were carried out, where different socio-technical issues came into focus in regard to the technical base and PD.

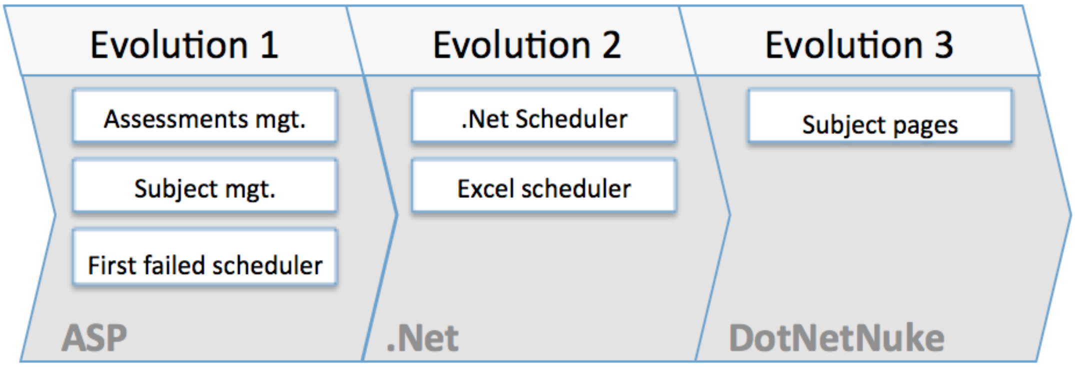


Figure 7.2. Indication of the projects and technical platforms of the three evolutions of the technical base of the action research

The focus here is not on the generic qualities of the technical base(s), but how it engages and matters in a socio-technical relation in the given developing settings. To prevent misunderstandings, it needs to be pointed out that the infrastructure evolution accounted for is not claimed to be attributed *only* to the technical base but how the technical base interacts with the social in the socio-technical development. In this thesis, the organizational process side of the infrastructure development is described in both the previous and following chapters. The account below though focuses on the role the technical base plays for the infrastructure and, especially, the development practices it affords.

To structure the account, the three improvements are described in three separate sub-sections below. For each improvement, it is indicated who/what was involved; the particular practice under scrutiny; the deliberations of improvements that were carried out; and finally the results. The three improvements are then evaluated together in the section to follow. In addition, and as described in the research approach chapter, the deliberations of the technical base also came to prompt other issues of relevance in regard to the infrastructure and PD. These are indicated in the text below and then further accounted for in the forthcoming chapters.

7.2.1 Improvement 1 – Talking about the technical base

The first development project that Jönsson (and later this researcher) was tasked with when entering the organization still kept within the realm of the academic administration but involved additional users in the development of an academic scheduler. The window for deliberations concerned: Could the shop floor IT management approach be opened up to involve additional users in development activities of a new academic scheduler?

Who was involved?: In addition to Wang who represented faculty interests in his capacity as both Vice-President (Academic) and a specialization professor, a primary stakeholder for the first versions of the scheduler was the faculty assistant Levy. Similar to the other early academic administration modules, the academic scheduling was carried out in Excel according to a template that Levy had taken part in developing and was organizationally influential in managing. For the scheduling, Levy was an established actor within the faculty, and he also had an established technical and organizational way of conducting the scheduling of courses and rooms together with faculty and staff members.

Understanding practice: Writing the code for the scheduler on the technical base in place for the academic administration was the most comprehensive programming undertaking at the time. Based on the original Microsoft's Active Server Pages (ASP), the development took place on a technical base, using a mix of in-house programming combined with code-snippets found on the web. Although an architecture influenced by object oriented methodology was in place that provided a basic framework for security and menu management, a substantial amount of programming had to be carried out. A layered, database driven architecture had to be customized where functionality and states were connected to an underlying (manually defined) database scheme.

Before this researcher was employed as an IT-professional, Jönsson had started to design a database scheme and basic interface structures for the scheduler using Levy's original Excel files as a template. When this researcher, about a year later, took over the development the completion of a first version was already delayed, and it took around six months to finalize a first version. It involved "information based" interfaces tied to the different subjects that were meant to support faculty for easy entering of teaching units.

The schedule module was, however, not adopted upon implementation, as expected. When introducing the new scheduler to Levy through a number of training sessions, he expressed that it was inadequate – from his point of view – as a replacement of spreadsheets for the creation of schedules. His main objection was that it did not satisfactorily take into account the "complex puzzle" that the scheduling process in reality entailed. To be able to book classrooms, coordinate subjects, and teaching units between different specializations, scheduling was not done subject by subject but rather holistically between all specializations. In addition, as professors usually submitted their schedules as notes in Excel and had many change requests, the ability to cut and paste in the spreadsheet program interface was a necessary desirable feature for Levy. The academic scheduler was thus not technically flawed per se, but that it was not useful given the particular work practice ecology with which it had to integrate.

Deliberation of improvements: The first PD improvements became how to talk to Wang and Levy and other stakeholders at WMU about the cause of the failed scheduler.

It gradually became visible in discussions between the IT-professionals Jönsson and this researcher that it was difficult to accommodate Wang's and Levy's combined development requests for the scheduler using the ASP technical base in place. Re-occurring entries made in the research diary were on the theme that the technical base was perceived to be too programming intensive. As Jönsson also stressed in the retrospective timeline workshop, for him this was the main reason that the scheduler became delayed: "[…] and I had told [Wang], that I have a lot to learn especially concerning ASP, but I was very keen. But then it was everything else […] I was sitting by the computer an hour or two, then interrupted, I had to sit down again […] there was [also] a lot of administrative work." For Jönsson, combining the level of technical programming necessary for the work approach that Wang had put in place at faculty – where IT-professionals worked closely with users not only in the process of design, but also with partly overlapping work areas – proved too much.

For this researcher, who had a professional background as a programmer, it was not ASP and the technical programming in-itself that was the issue. The program intensive nature of the technical base was, however, still a concern, as it was too time consuming to develop prototypes and create proof of concepts to illustrate different solutions and then turn them around to production code within an affordable timeframe. Based on a set of workshops where more time was committed to work with Levy developing paper-mock-ups together with vertical and horizontal prototypes, a graphical planning was envisioned. However, there was not enough project time available to take it to a satisfactory operational state.

From the user's perspective, it was also an issue how the technical base could both enable and constrain development, but which was difficult to comprehend: Levy later expressed, for example: "You were so pre-occupied with your technical stuff […] you were just not listening to me […] that was not going to work", which was an indirect consequence of the time-consuming nature of the technical platform. Another statement made by Wang was later used in the IT-coordination group to indicate the challenges for users to engage in discussions concerning these matters: "Just put it into the database!" Though he himself was an IT-savy end-user developer, understanding the "fuzz" of how a technical base could have affordances on the development carried out was still not straight-forward.

The first improvement of the action research deliberations addressed how to make these experiences available for reflection in a meaningful way beyond only the IT-professionals. One explicit boundary object to accomplish this was the first reflection paper, which was based on the idea of framing the technical base issue as a story grounded in the users experiences, became : Using the issues Levy encountered during the implementation of the first version of the scheduler (see the first paragraph in Figure 1) as an explicit instance, this researcher started gathering experiences as a base for reflecting on opening up the "PD one-to-one" development mode to involve more users. The first reflection paper was used as a tangible means to support discussions between the IT-professionals Jönsson and this researcher and the users Wang and Levy, regarding the technical base. In addition, it was used as one of the early inputs to the workings of the newly formed IT-coordination group, gathering WMU's three IT-professionals under coordination of the Head of Information.

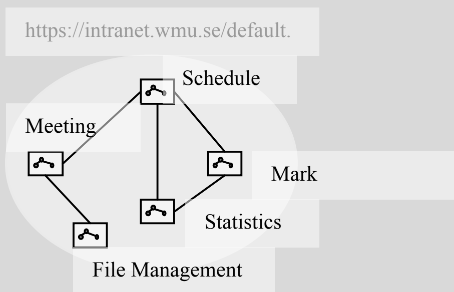
Result of deliberated improvements: The first improvement did not actually involve a change of the technical base per se. The overall contribution of the deliberations were to connect the technical base with PD matters. Hence, organizationally the technical base was not only a concern for IT-professionals, but also had tangible implications of how software support could be developed in local shop floor projects with users such as Wang and Levy. By bringing the reflection into the IT-coordination group, this realization could be extended beyond just the scheduler development. The premise underpinning the IT-coordination group was that increasingly development projects would involve functionality intended to intertwine in close ways with already established work practices and tools of a growing number of users also outside the faculty. The analysis shows that, in this way, understanding the implications of the technical base as more than a pure technical matter became a foundation for the nature of the deliberation of further improvements accounted for below. The early improvements enabled through the first reflection papers to better talk about technical infrastructure between stakeholders, also contributed to more extended attention of such matters described in chapter 8.

Reflection Paper 1 – Thoughts on the faculty IT activities (8 pages)

The issues presented in this text are basically nothing new, and have been part of the ongoing working process. This has involved informal hallway discussions up to and including scheduled meetings. The contribution of putting them on paper is hopefully to provide additional transparency and legitimacy

[…]

<u>Software Development platform</u>



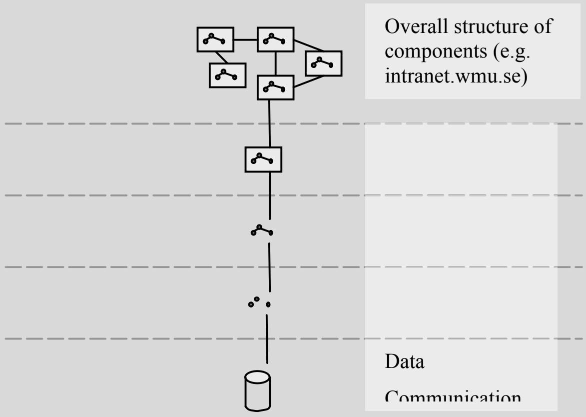
The intranet is currently running on a script based programming platform called Active Server Pages (ASP). Due to various constraints, and an increasing demand from the university of more sophisticated application functionality, this platform is becoming increasingly cumbersome and inefficient […] During my and my predecessor's time, most functionality has been developed within the framework of the web based intranet. An ongoing issue has been, for example, the duplicate operations that Levy performs when inputting the schedule. The extra work stems from that he prefers to do a draft in Excel before entering the data in the web based interface that is available today […]

[in final version made available in phase 2]
Preliminary investigations carried out show that the .NET platform may utilize the programming model we currently have. And as ASP.NET is an upgrade of ASP there is a certain degree of backward compatibility, which makes a migration of existent functionality not impossible, but at most time consuming. […]With the .NET platform interfaces are also opened up to other applications in ways that did not exist with ASP (together with COM components in Visual Basic). This enables us to bring functionality directly into the applications that are used by our user community in addition to the intranet.
[…]

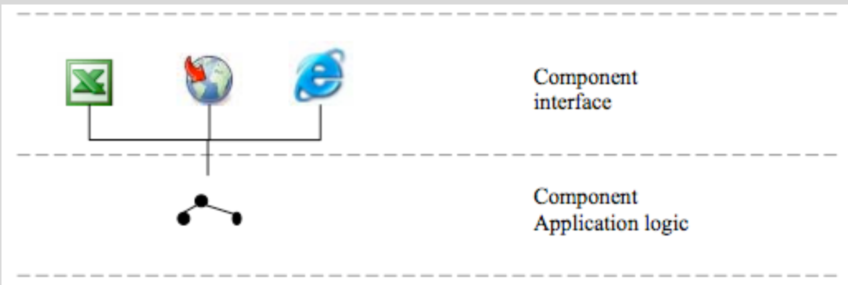
Project Management
The people that are involved in different contexts of both the development process and the continuous usage of software are most often both engaged and knowledgeable about their various work domains. They know how it works and have a lot of ideas of how it is possible to improve thing […] this extends from the Vice President, academic, [Wang] to the different faculty assistants. The knowledge needed to design and implement software is thus best developed through active cooperation between the day to day users and the software developers […] This in order to achieve the best fit between technology and the way that people actually work […]The challenge here becomes to slowly and steadily put these ideas into practice […]

Empirical excerpt 7.1. Excerpts from the final version of the first reflection paper. Connecting technical base and organizational IT process considerations

Reflection Paper 2 - Thoughts on the faculty IT activities: revisited (5 pages)
In the previous reflection paper, [we] shared thoughts on the faculty IT development activities. The paper, based upon the current situation, focused on how the different parts of the development activities could be improved […] The .Net framework is now in place on the production web server and the intranet is migrated to the new environment […]A main difference to the old design is that the number of independent layers has increased.



One important benefit of this is that the user interface and the component application logic are fully separated. In addition to improving the structure and readability of the code this means that the functionality can be reused in different contexts, as shown in figure […] The potential of this is quite extensive. For example, an application has already been developed that enables an Excel template of the schedule to be directly exported into the intranet database. This has already saved a considerable amount of time when processing the schedule for semester 3 […]



Empirical excerpt 7.2 from the second reflection paper - following up on technical base revision

7.2.2 Improvement 2 – Improving "use in design" through revising the technical base

The overall questions that framed both the understanding of existing practice and the deliberations of improvements of the second round of improvements became how the technical base could be improved to involve additional users in development efforts.

Who was involved: In the second round of improvements, both the IT-professionals Jönsson and this researcher became active as technical developers but for different application domains; this researcher continued to work with Wang and Levy, but also with additional faculty members to enhance the academic scheduler. Jönsson left the academic administration development and started to work with staff members from faculty and library to develop research project portals, an alumni portal and a library portal.

Understanding practice: As field notes and audio recordings show, what triggered the action research deliberations were the affordances of a new ASP.NET technical base. To develop research portals, a library portal, and alumni portal, Jönsson started to explore how to conduct technical development in the emerging fourth generation technical bases. A feature with a fourth generation technical base is that it does not necessarily require traditional programming to produce an outcome. A complete technical product can be implemented through administration and configuration of an existing framework and existing software modules. Jönsson started to use a technical base built on ASP.NET 2.0 called SOOP for the requested development projects. He expressed the benefits of SOOP and fourth generation technical bases as "the thing was that we got all this functionality in [fourth generation] tools when it came [to] the launch of a new portal, …another benefit was the framework updates that one received through the CMS tool's own updates."

These experiences led to that the users proposed functions using the fourth generation technical base for the academic administration too. Doing so also had the potential to carry several synergies both in terms of shared functionality and technical development capabilities. One such example put forward was a common document management function where documents could be indexed and shared across portals.

Deliberation of improvements: However, it was unclear whether the fourth generation platforms were able to support more complex functionality that needed to be tailored to the specific work practices that would be necessary for the academic administration modules.

In the second evolution, the first reflection paper was complemented by a second (Figure 2) that followed up on technical infrastructure dimensions about the revision of the academic administration technical base. As a result of the expanded technical development, the reflection paper's role as boundary object was expanded both on the IT-professional's side and on the user's side. In addition, prototypes were developed to show proof of concepts of SOOP and to aid the further improvements of the academic administration supported the deliberations with the users.

With these discussions, an understanding was negotiated that (1) it was not certain that the existing tailored academic administration functionality could sustain in SOOP and (2) that a gradual migration from ASP to ASP.NET 2.0 could still carry some of the benefits of fourth generation technical bases, such as rapid prototyping:

1) As a result, the internal evaluation showed that the then emerging versions of fourth generation technical bases were often built on a particular development instance that was then adopted for generic development / configuration to various degrees. SOOP was, for example, built based on a forum development instance with security, menu, and pages management, among others. It had then been architecturally expanded to include document and picture management, etc. This was satisfactory for the research project portals, the alumni portal, and the library portal. However, it could not be assured that it was sufficient to technically maintain the academic administration functionality.
2) The reflection papers and prototypes also showed that the ASP technical base of the academic administration could be migrated to ASP.NET 2.0 and that this could carry a number of benefits to the development approach, such as better built-in separation of interface and programming logic, better API's toward other programming environments, event driven programming, and availability of ready-made programming components.

Result of deliberated improvements: The first improvement grounded a practice of talking about the technical base that was continued in the second round of improvements. Based on these reflections, a new technical base that improved the possibilities to work with PD in local projects could be implemented.

Using inspiration from the architecture of the fourth generation technical platform SOOP, the existing tailor developed academic administration functionality was merged into a more distinct multi-tier and component-based architecture in ASP.NET 2.0 (see first picture in Figure 2). Two of the prototypes used as proof of concept were also finalized to operational versions and show how PD was furthered in two different ways in involving users in design:

- Based on Levy's request, using the improved API's, a component interface was implemented to enable the import of Excel data to the scheduler. A bridge was thereby created to the old established way of planning the schedule. Although it still involved trade-offs, it meant that the schedule could continue to be worked out in an Excel spreadsheet. In addition, the schedule could be made available in other environments, such as the web, as well as the University's email and calendar tools called GroupWise. The new technical base thereby made it possible to better accommodate users' particular design requests.
- One ready-made programming component that started to be used was the DataGrid that can output a complex and configurable table structure based on a single DataSource assignment and property configurations. The DataGrid is an example of how it technically became easier to showcase functionality through prototyping. In this sense, it shows how the technical base freed up time to cooperate with users. Using these components, an improved interface was created to view and navigate multiple and combined schedules at the same time.

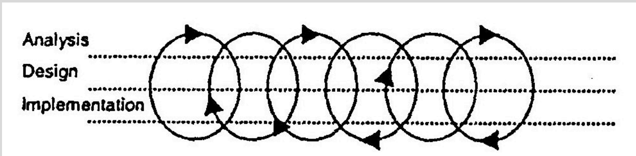
Reflection Paper 3 – The designers? That's us! (5 pages)

WMU has chosen a model of internal software development where the IT function is placed in the middle of the faculty organization.

The former ASP based platform required a focus on the software to be produced in itself and not its usage. The consideration that had to be given to the coding process was substantial. In addition, the structure used made it difficult per se to easily involve users in, for example, interface design, which is one way to open up and get feedback on the development process

The title of this reflection paper is a quote from what is traditionally thought of as the users of a software development project that was reported on by an on-going research project in Dittrich, Eriksén, and Hansson (2002). The purpose of using it is that it expands the idea of who is designing software from only the software developers to also include other stakeholders, as, for example, what is traditional thought of as end-

user.
A first and important step in this process is the new .Net platform which has been gradually been implemented. By making new tools available to the software developers, it opened up the development process and improved the opportunities to work closely together to the users […] one of the main advantages thereof is that an integrated perspective on analysis, design, and implementation can be achieved.



The above figure displays a basic model of an evolutionary development process that makes it possible to achieve active collaboration during development

Empirical excerpt 7.3 from the third reflection paper. Focusing on organizational IT process improvements. Picture from Christensen, Crabtree, and Damm, *(1998)*

7.2.3 Improvement 3 – Revising the technical base to enable "design in use"

As a consequence of the previous improvements, more and more organizational actors became involved in the PD practices around the academic administration and the research portal. The question was how a particular fourth generation technical base could be used from an organizational IT process point of view to (1) better involve an increasing number of users in design and (2) how the users could themselves carry out design activities as End-User Developers (in accordance with the original shop floor development model described in section 7.1).

Who was involved: The revision of the academic administration functionality described in the next section featured an increasing number of users actively involved in design processes. Workshops during the course of the action research were carried out with the shop floor people: Professor Wang, Levy, lecturer Hansson, Professor Mercier, Professor Nakamura, and Associate Professor Schulz.

Likewise, the research project portals, alumni portal, and library portal were complemented with the external website which also required participation from administrative staff and the president's office. In addition, the technical base was now negotiated not only between departmental users in local shop floor projects but also the IT-professionals in the IT-coordination group. Increasingly, an IT-steering committee with cross-departmental representatives was involved in strategic organizational and technical infrastructure matters. This also links to the second Cooperative Method Developers research cycle of improved organizational IT management accounted for in chapter 9.

Understanding practice: By now, in general, the functional capabilities of the available fourth generation technical bases had expanded and their architecture - also to aid in-house tailored development - had matured. Following the implementation of SOOP, Jönsson continued to investigate different kinds of fourth generation technical bases. The primary option put on the table was DotNetNuke, which At the time of the action research was the most widely implemented ASP.NET 2.0 based open source Content Management System

At the same time, the IT professionals became more and more aware of the shop floor IT management based on PD. Different practices of PD were investigated, as presented in the previous chapter.

Deliberation of improvements: Similar to SOOP, DotNetNuke is open source and based on ASP.NET 2.0. However, it is not fundamentally designed towards any particular area of application. An extensive catalogue of modules is available through the surrounding development community, both for free and for purchase (www.snowcovered.com). Of importance was also that an established framework existed for tailored development of modules with documented API's of how interaction could take place with the wider portal framework functionality (to get access to user, role, menu management, etc.). A final and key benefit was built-in support for end-user administration and development that enabled users to carry out design and update information on portals and websites themselves. Again technical experiments were implemented to make sure that the already established functionality could be carried over to the new technical base.

A third reflection paper created a bridge between the technical infrastructure and the work that the IT-professionals conducted with the technical base with early deliberations of improvements to the organizational IT management process. The evolutionary software development cycle in Empirical excerpt 7.3 was, for example, used to spur a discussion of how the type of rapid prototyping already earlier enabled could continue to be improved with DotNetNuke. In addition, metaphors such as "The designers? That's us!" became influential in showing a path forward for how users could adopt functionality to fit their own need as End-User Developers.

To deliberate on how these objectives could be realized in concrete development projects, the development of a project management approach based on the MUST framework (Bødker, Kensing, & Simonsen., 2004) was prompted (further described in chapter 9), which extended the design dimensions of the shop floor agile development implementation approach already in place.

Result of deliberated improvements: An extensive development review of the academic administration functionality made possible with the migration to DotNetNuke was to evaluate the implemented improvements. Based on the local implementation of MUST, seven in-depth analysis sessions were conducted where Levy was involved together with different constellations of five other faculty staff members. The outcome of the in-depth analysis was then used as a base for several innovation workshops where eight overall story cards were created to structure the design and implementation. The development was coordinated both with the IT-coordination group and the IT-steering committee. The technical base DotNetNuke enabled PD between faculty and support staff as follows:

- Using a set of three custom academic administration components that are based on the functionality carried over from previous technical bases, Levy defines a subject and schedule including information, such as professors' in-charge, number of credits, and participating students. The schedule can be entered both in a web interface and through excel.
- Based on Levy's definition, a generic subject page is created in the DotNetNuke framework and access settings are automatically assigned to the professors and students connected to the course.
- In addition, a mix of customized and out of the box modules related to academic administration automatically is added to the subject page. These components are then administered and populated with information by the professors themselves. They include assessments (custom), e-documents (ready-made), and syllabi (ready-made). A professor is also able to add additional components to the subject page to support her/his teaching activities, such as forums, picture galleries, and project management components.

Overall, the academic administration showed that the DotNetNuke technical base was capable to both support rapid prototyping and support users to be part of in-design activities together with IT-professionals and – provided a careful design – support users to carry out adaptation and development themselves.

There are, however, also instances where the implications of the technical base have not led to improvements. One example of this is the document management module that was referenced as one of the examples of potential synergies already in evolution 2. For this functionality, a mature out of the box module existed that was already used in the research portals. Staff in administration and registry now prompted the IT-professionals on multiple occasions as to why it was not possible to send out notification any longer when uploading documents. In the same way, faculty had to come to terms with the fact that they could not set individual write and read permissions on files. In this particular case, it was not possible to find a fully satisfactory solution, such as a work around or a tweak to the standard nature of the module or replacing it for part of the infrastructure.

7.2.4 Summary

The research reported above shows how the technical base as part of the socio-technical infrastructure both enables and constrains design collaborations in the local projects. It is also shown how users and IT-Professionals can engage in PD of the technical base.

- The first evolution established the use of reflection papers to involve users not only in the design of specific functionality but also in deliberate possibilities and constraints of the existing technical base, e.g., when involving more users and developing more comprehensive functionality. This issue prompted more action research accounted for in chapter 8.
- The migration to the ASP.NET 2.0 technical base in the second evolution improved the user developer collaboration by providing time and capabilities to work with additional users.
- The third evolution showed how the change to the DotNetnuke platform opened up for involving more and more users as End-User Developers. The additional requirements that this put on organizational IT management is further accounted for in chapter 9.

Moreover, other dimensions of how the technical base influenced infrastructure development became visible. In the case of the document management function, decisions made in one project, for example, influenced other parts of the infrastructure as well.

7.3 CMD 1 phase 3: Evaluating the role of the technical base in infrastructure development

Whereas the previous section provided a rich account of several cycles of deliberation and change of the technical base at the WMU, indicating how the choice of the technical base not only influenced the specific functionality but also the possibilities for PD and PD, this section details several aspects of this realization. Through the research it is understood that infrastructure standardization and local development *need* and *can* be developed hand-in-hand. It is shown how, in this case, end-users triggered infrastructure development both in a conflict-full unplanned manner and through planned and moderated deliberation. It is then highlighted how the technical base influenced the possibilities for PD and that it needs to be carefully considered when intending to sustain PD. The fourth subsection then extends the need for organization-wide participatory deliberation of the technical base decisions also on jointly used modules, and, last but not least, the need for further development of methods and tools to do such deliberation in a participatory manner is argued.

7.3.1 Infrastructure technical base affording local design and vice versa

The case shows how the technical base plays an active part in the dynamics between a common infrastructure enabling on the one side and reacting to local development on the other:

The first technical infrastructure with ASP as a technical base enabled custom solutions in the scheduler project and was underpinned by Wang getting an understanding for databases through excel and being inspired by web 2.0 development. This very same technical base then negatively influenced the attempts to open up the PD one-to-one development mode in evolution one to encompass the faculty assistant Levy's requirements. In evolution two, the local development with the technical base SOOP enabled new architectural options that gave more time and flexibility to consciously deploy PD tools and techniques. Finally, in evolution three, the fourth generation technology enabled and pushed new ways of user involvements in form of PD in local shop floor projects.

The standard set by the technical base and the infrastructure developed on top of it thereby affords local technical development and the evolution of the infrastructure. At the same time, emerging technical options that are situated in local development push the development of the infrastructure and – in the end – provoked changes in the technical base.

7.3.2 Users triggering infrastructure development

The empirical material shows how users are not only users of the infrastructure, but also engage in the design deliberations of the technical base. As reported, the users were among the actors arguing for changing the technical base to a more flexible solution that allowed the easier development of interfaces to other programs, such as excel and word editors. This 'shop floor'-influence of the infrastructure happened both in planned and unplanned ways:

An example of an unplanned influence that in situ showed as conflicts is Levy's critique during the rollout of the first version of the scheduler which was unexpected and was perceived to be disruptive by the involved IT professionals and main stakeholders. This, however, does not mean that the criticism was illegitimate. As it led to a substantial technical investigation, it made visible that the technical development means available through the existing infrastructure were simply not appropriate to create a useful solution according to his requirements. From the individual project perspective, the scheduler was a failure. The inadequacies of the developed pieces of functionality communicated by Levy, however, made their way into the reflection papers and, thereby, became input to the discussions of the IT-coordination group and IT-steering committee regarding the change of the technical base and thus to the direction of the infrastructure development.

An example of planned influence is the participation in exploring new possibilities in a PD way during the second round of improvements with the help of the reflection papers. In conjunction with the launch of the second revision of the technical base, a decision had to be made whether to change to a fourth generation technical base at this point or incrementally upgrade from ASP to ASP.NET and develop a better multi-tier and component based architecture. To this end, it was possible for the users to test and give feedback on how they envisioned the continued development grounded in the prototypes tied to the reflection papers. Similarly, in the final improvements in regard to the technical base, the experience with some of the capabilities, such as the usage of ready-made components, influenced the succeeding transformation to the fourth generation platform.

Be it unplanned or planned, the examples here have shown how users successfully influenced the deliberation of the technical base and development of the infrastructure based on it.

7.3.3 The technical base frames the design practices

The technical base not only enables and constrains the functionality of the infrastructure and the programs integrated into it, but also that it influences the design processes. In the traditional perspective, PD is an input to the development process, which, in turn, influences the design of the product that then frames the possible ways of using the software. The empirical data show how the technical base once implemented enables and constrains different collaboration patterns between IT-professionals and users. The first technical base enabled the development of highly custom pieces of functionality according to Wang's specific requests. In important ways, this defined the nature of the close development cooperation that was formed between Wang and the IT-professionals. The fourth generation technical base, DotNetNuke, still enabled tailored development but geared the IT-professionals more towards combining own and ready-made components developed by third party vendors. This made it possible to quickly showcase pieces of functionality. It also opened up opportunities for end-users to configure and develop functionality themselves. The components, in turn, became part of (and dependent on) the DotNetNuke framework. This provided more flexibility both for integrating local practices and for end users to contribute to the development of the infrastructure. The technical base thus not only affected the design space, but also influenced how the working relationships of technology production and use could evolve.

7.3.4 Is the holy grail of WEB 2.0 the answer to our participatory prayers?

The developing capabilities of Web 2.0 carry important opportunities. However, the argumentation in the previous section of how the technical base affords local development and frames PD practices does not warrant a naive praise of Web 2.0 as participatory technologies.

The argumentation of the previous section- that different technologies have different affordances - can accordingly also be explicated with regard to Web 2.0: ASP, ASP.NET 2.0, and DotNetNuke, as all fall within the classification of Web 2.0 technologies. However, even though they, as Microsoft technologies, showed certain compatibility, they also have clear differences. ASP.NET 2.0 is not only an upgrade of ASP, but implements substantial architectural differences in the sense of event driven programming and programming components such as the DataGrid. Even though the DotNetNuke technical base was based on ASP.NET 2.0, the nature of tailored development became fundamentally different; it now takes place as module development within a framed shell in which the own models interact with other ready-made components that pose additional constraints.

The example shows that Web 2.0 technologies are both enabling and constraining. High-fidelity prototypes and mash-ups have added new capabilities for users to participate in the development of software support. Nevertheless, they have also added constraints to custom development, thus making it more difficult to communicate the opportunities and limitations of tailoring and configuration.

7.3.5 The technical infrastructure links the IT support of separate work practice ecologies

The document management module and its usage in the academic administration provide another insight into the reflection of the role of the technical base in PD. The document management module was selected and implemented to support a number of functions before being used for the academic administration. In these cases, it successfully supported the work practices. When working with the academic administration, the former decisions turned into a straight jacket. The technical base linked software support for different local work practices. The relation between local decisions on specific pieces of software support needs to consider implications on the whole infrastructure.

The reader might argue that one could just use another document management module or write a new one the different local projects, thus making technical infrastructure considerations abundant. To this end, there is an abundance of both proprietary and open source document management modules available for the DotNetNuke technical base. However, there are a number of arguments for the choice of *one* document management module for the whole infrastructure. By having the same document management module, it was possible to integrate all the uploaded documents in common dashboards, for example. A uniform way to manage documents, administration also became more intuitive for the users – knowing how to administrate documents in one portal, they knew them all. The implication is that choices with respect to one part of the infrastructure influence the design in other parts. In this case, the document management function could not be optimized for the academic administration.

Optimally, a satisfactory "holistic" technical solution seeing to different needs should have been put in place, which would have had to be negotiated outside the individual project. However, this would have mandated that users understand and engage in deliberations regarding the technical base outside their particular "work infrastructure" - the parts of the infrastructure they actually use (Pipek & Wulf, 2009). Forums such as the IT-coordination group and IT-steering committee are examples of possible forum at WMU to negotiate the tradeoffs of the application of the particular modules. To do so, meaningful ways to discuss technical infrastructures with users is needed.

7.3.6 Deliberating technical base decision with domain experts

If the technical base is not black boxed, the challenge is to find meaningful boundary objects to mediate the understanding of the technical base. The approach in this study did not pursue the route of traditional IS management, where a substantial body of hierarchical modeling documentation constitutes the boundary object for technical base deliberations. The rationale behind mediating the technical base deliberations using, for example, the reflection papers aims at involving users in the deliberation of the IT infrastructure for the whole organization.

Using short summaries of incidents in current work and development practices, thus providing a vision for future change and being developed iteratively, the reflection papers were intended to work as "reminders" for discussions and deliberations in the sense of Kyng's (1995) discussion of the representational qualities of artifacts. In the current form, they are, however, more to be regarded as a probe of how to represent infrastructure for PD, rather than a final solution.

What the reflection papers showed at that stage is that it is possible to create meaningful representations of technical infrastructure development that raises intentionally selected qualities for deliberation together with non-IT professionals.

However, there are still unanswered questions that warranted more attention: What did the users actually understand? Did they understand the infrastructure deliberations or did they understand that the IT-professionals understood them? In the case of a weighting towards the latter end of the continuum, it is still the IT-professionals who are the stronger actors. This is an issue that is revisited in the action research deliberations in chapter 8 concerning how people on the shop floor can talk about infrastructure.

7.4 Summary

The results of the action research reported in this chapter contribute to understanding the technical infrastructure dimension of sustaining PD in the organization. The analysis of changes to the technical base of the action research towards a PD strategy made visible the role of the technical base in such endeavors: It is shown that the technical base not only constraints and enables the final design, but also the design and development process and the possibility to involve heterogeneous stakeholders in its development.

The technical base on which the infrastructure is built is crucial in providing the flexibility to both accommodate heterogeneous and changing work practices. A contribution of this study is to shed light on infrastructuring where technical design meets use on the shop floor; the dynamics of the technical base of an infrastructure both defines and is defined by local development activities; something that became of increased relevance to the shop floor IT management accounted for in the chapter, as well as in the previous chapter. The relationship between the infrastructure and the local development, though, does not need to be polarized. In this case the infrastructure enabled more supportive functionality.

Moreover, the possibility of including heterogeneous stakeholders and opening up for PD depended on the selection of the technical base. The advancing capabilities of Web 2.0 technologies were promising in the development of the academic administration infrastructure and in PD by facilitating the creation of high-fidelity prototypes. However, there is "nothing like a free lunch." Although it is easy to get carried away by the capabilities of Web 2.0 technologies, they need to be considered from a technical infrastructure perspective to be useful in sustainable development beyond the local development project.

Thinking about infrastructuring in terms not only of meta-design and anticipating but also of organizing - or the dynamics between local design and infrastructure evolution might provide a frame for handling design on various levels. Seen from both a local project and organizational infrastructure standpoint, it becomes necessary to consider how flexibility with respect to PD comes at a certain expense, e.g., design decisions regarding unrelated parts of the infrastructure actually influence the design space in later projects. However, this also requires PD beyond local development. In this way, the outcomes of the research regarding the technical base have prompted how shop floor people can organizationally manage infrastructure development, as well as PD tools and techniques that span local shop floor development constituencies. The former issue is the topic of chapter 9. In addition, current PD techniques and tools give limited support to negotiate such influences, as they are remote to the local situated design. The experiments with reflection papers using concrete development situations and technical experiments to explain the affordances of different technologies are only a start in this direction. The appropriation of PD tools and techniques for infrastructure development is further developed in chapter 8.

8. Let's talk about infrastructure! – In a Participatory Design way

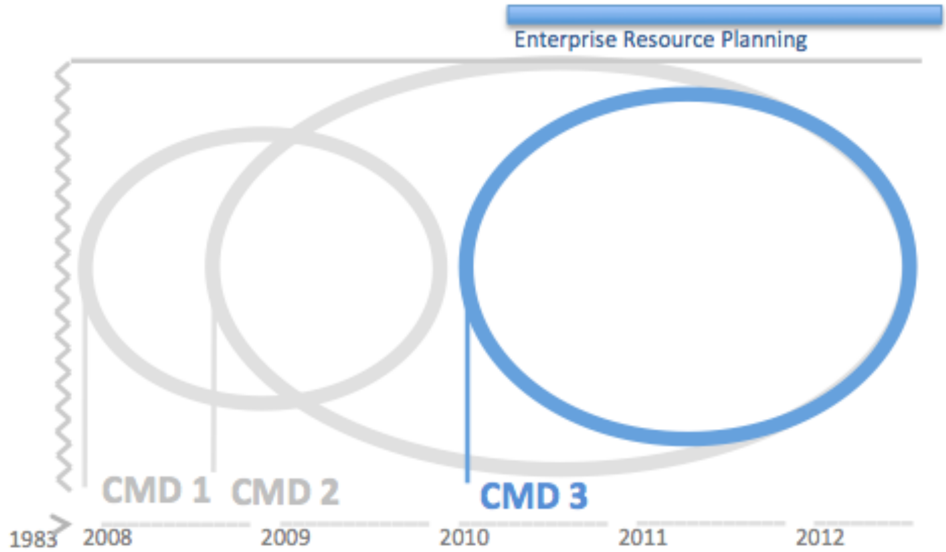


Figure 8.1. The third Cooperative Method Development action research cycle illustrating how the shop floor development constituencies together with IT-professionals and vendor representatives could understand and deliberate the design of new ERP system WMU (see complete timeline in chapter 4).

This chapter is about the use of Participatory Design (PD) tools and techniques for infrastructure design. The chapter reports on the third Cooperative Method Development (CMD) research cycle where the appropriation of working PD tools and techniques in-itself was the focus of the deliberations. The need of improved PD tools and techniques was realized already during the first CMD research cycle. The research results and the homegrown approach of reflection papers were not only about improvements of the participatory qualities of the technical base, but also pointed to additional research needs regarding PD tools and techniques, as well as participatory organizational IT management. As the above figure indicates, the third CMD research cycle, the main focus of this chapter, targets improvements to PD tools and techniques. The third CMD research cycle is, in turn, part of the second initiated CMD research cycle that targets the organizational IT management, which is reported on in the next chapter.

A design project about an integrated ERP system was used as a case for the deliberations about PD tools and techniques in infrastructure development. The ERP design project involved all the established shop floor development constituencies that have been reported on so far: the registry and administration shop floor development constituencies (chapter 5 and 6) and faculty shop floor development constituency (chapter 7). In addition, the development- and use constituencies of the finance and HR department were part of the ERP design project. Given the established shop floor IT management approach, a central aim was to maintain a strong user influence in new design ideas put forward. A challenge to a PD approach posed was to replace the technical base of different shop floor development constituencies with one new integrated system. Working local software support not only had to be re-designed, but also integrated with other technical and organizational systems. In addition, previous local development- and use constituencies in the academic faculty, academic registry, finance, and HR departments had to 'in part' be re-conceptualized as the raison d'être and mode of operandi of how they worked with and how the designed software support was changed.

This chapter accounts for a number of dimensions of how common working PD tools and techniques add value to integrated technical and organizational infrastructure design. PD tools and techniques have over the years proven their value to support users on the shop floor and IT-professionals to design user-centered IT systems, but have predominantly been applied in a local project setting. The empirical results of this chapter show how such common PD tools and techniques can add value also outside their commonly documented function in regard to integrated technical and organizational infrastructure deliberations. The contribution of this chapter is to account for new usefulness dimensions and challenges in applying PD tools and techniques when used in an integrated infrastructure setting.

In the following section 8.1, the background and need of a PD approach to tools and techniques in the ERP project are described. Their application in this research study is then described in section 8.2. The final section describes and evaluates how these tools and techniques – both individually and in combination - provide value beyond their documented function in the design of an integrated technical and organizational infrastructure.

8.1 CMD 3 phase 1: A need to talk about integrated technical and organizational infrastructure

The ERP design project was governed by 4 + 1 overall strategic requirements that were put forward by the IT-steering committee and the management of the university: (1) the four functions of Finance, Academics, Human Resources, and Research Project Management were to be targeted; (2) the IT-support and work practices these four areas were to be better integrated; (3) the solution was to be standardized (in the sense of a reduced dependence on internally tailored technical systems); (4) the solution was to build on best practices in the industry of university ERP systems. The (+1) requirement recognized WMU's shop floor development approach of domain experts being able not only to use but also configure technical solutions. This approach was to, if possible, be maintained and cultivated. This connects to the outcome and evaluation of the deliberations of an organizational IT management from the shop floor reported in the following chapter.

Also from a new management point of view there was an aspiration to do something "new" in regard to ERP development. A new Head of IT and Administration, Andersson, with a background in project management in banking as well as in a United Nations office brought extensive experience of ERP development with him. However, when the best practices of the development approach that he previously had used was scrutinized, given the modos operandi at WMU, it was judged that it was not suitable because too much of the project time would have to be spent on formal documentation models alone. This would have had implications on the time available to work directly with end-users in line with the shop floor IT management approach at WMU.

Based on the 4 + 1 requirements, the ERP project was to (1) investigate what potential ERP solutions were out there, (2) to solicit requirements from the different departments, and (3) to present a design proposal with a business case.

8.1.1 A focus on integrated academic function issues

To make the issues tangible for the reader and how the PD tools and techniques were applied in practice, the empirical account and evaluation put forward in this chapter primarily focus on the academic functions of the ERP design project. These were the most comprehensive and complex functions of deliberation in the project. At WMU, as at other universities, the academic functions are a core area. This has also been related in the previous chapters. The academic functions include the need for system support in areas such as student, subject and mark management, and reporting.

In previous chapters it has been described how from the beginning these were organized separately between the registry and faculty department, and how there were synergies to be gained from integrating them. This entailed two separate technical support systems to be in place, designed and implemented by different local shop floor constituencies with different development approaches and dynamics. The registry system was the most comprehensive system support in place at the university. As was described in chapter 5 and 6, it was also technically developed by the Registrar himself in close collaboration with his staff – which comes to triangulate how their original shop floor development approach was perceived, compared to the unfolding in the ERP design project used in evaluation here. For the faculty system, this researcher was the primary technical developer during the action research of the thesis. Chapter 7 and 9 report on how a shop floor development constituency consisting of faculty, assistants, and this researcher collaborated in evolutionary design and implementation activities.

A development throughout the action research cycles in this thesis was that, for many functions, separate shop floor development constituencies were increasingly dependent on integrated workflows. This was also true for the registry and faculty, and was one of the primary challenges of the ERP design project; for example, upon the registry department receiving a student enrollment application and the student is accepted by the faculty and funding is secured, there are numerous reports and forms that need to be produced to support meetings and decisions within and between members of the respective departments. The reason to take additional steps towards a integrated solution within the ERP design project was that there were still many manual interactions - something that was recognized by both faculty and registry staff as being time consuming and error prone.

However, the main advocate of a new integrated solution was the faculty. Primarily, faculty wanted to have reporting and input interfaces directly to data in the registry system. On the other hand, this was also the source of an explicit long-term tension between the departments. Due to of data integrity and student privacy reasons, the registry department historically had, for example, been hesitant to let faculty have direct access to the registry data. Understanding the nature of this problem and being aware of the extent to which separate sub-systems had to be maintained were the primary challenges underpinning the integration of academic functions in the ERP design project.

8.1.2 Participatory Design tools and techniques used

As has already been shown in the first action research cycles about PD of a shared technical base reported in chapter 7, being able to talk about infrastructure between different stakeholders was an important realization in addition to the participatory improvements to the technical base. To this end, a homegrown approach called reflection papers based on Beck and Andres's (2004) notion of story cards was developed. As the number of shop floor development constituencies involved in the design and targeted dimensions of technical and organizational infrastructure increased during the second initiated CMD research cycle, so did the requirements on working PD tools and techniques. In the second CMD research cycle, managing organizational IT is in focus (chapter 9). Bødker, Kensing, & Simonsen's (2004) MUST method for project management was used as a base for deliberations. However, the knowledge framework as part of MUST approach also started to be used to improve the systematic application of common PD tools and techniques in the local project.

In the third CMD research cycle reported on here, a number of PD tools and techniques in the MUST knowledge framework were appropriated and used by the ERP project for infrastructure design, beyond the local project. These PD tools and techniques were selected during the ERP design project to answer to situated needs, and were based on discussions primarily in the design project group and the IT-steering committee. The tools and techniques selected were functional analysis, company visit, participatory observations, story card summaries, IT-professionals overtaking work functions, workshops, experimenting with prototypes, and developing scenarios. The function as documented in the MUST knowledge framework is described in the excerpt below. In the following sections 8.2 and 8.3, the application in the ERP project is accounted for and evaluated.

Overview of the Participatory Design tools and techniques used in the deliberations

In accordance to the MUST knowledge framework, the PD tools and techniques used in this research are described below in reference to the knowledge area and the principal PD function to which they relate. The MUST knowledge framework positions PD tools and techniques in three knowledge areas where users and IT-professionals need to generate knowledge in a design project (Bødker et al., 2004, p. 62). This is accomplished through a mutual learning process where users need to acquire knowledge about technical options, IT-professionals need to acquire knowledge about the users' present work practices, and they together need to deliberate on new IT usage. Within the three areas, knowledge needs to be gained both on an abstract (e.g., documentation and functional analysis) and concrete (observations and prototypes) level:

	Users' present work practices	**New IT usage**	**Technical Options**
Abstract knowledge	A. Relevant descriptions of users' present work practices	B. Design visions and proposals	C. Overview of technological options
Concrete experience	D. Concrete experience with users' present work practices	E. Concrete experience with new IT usage	F. Concrete experience with technological options

The core PD principles express the essence of what the sum of the PD tools and techniques positioned in the knowledge framework should set out to achieve (Bødker et al., 2004, p. 53; 2004): (1) the principle of a coherent vision, (2) the principle of genuine user participation, (3) the principle of firsthand experience with work practices, and (4) the principle of anchoring visions.

Functional analysis	
Knowledge areas: A	Principles: 1, 4
Functional analysis is a technique for analyzing work functions. It is positioned as a comparably management-oriented technique. Functional analysis starts from the company's overall business strategy. The aim of the analysis is to identify the work functions upon which the design project should focus (p. 208-209). As a starting point for a functional analysis, a work system must be identified and defined. The functional analysis is then performed by alternating among interviews, document	

analysis, and functional modeling (p. 238- 239).

Company visit	
Knowledge areas: C, F	Principles: 1, 2, 4
Company visits are a technique for gaining ideas and experiences regarding the types of IT usage that is being considered in the design project. A visit to another company using the relevant new IT usage may generate a wealth of ideas for the design project. It enables the IT usage's functionality, required infrastructure, or work organization to be studied. A valuable aspect of a company visit is the opportunity it affords for listening to, and discussing, experiences with implementing and using the IT systems in question. Whenever possible, observation of the system in use or hands-on testing of the system will provide more concrete experience (knowledge area F). Company visits may be set up via personal contacts, by cold calls to other companies, or via the supplier of the system that is being considered (p. 214).	

Review	
Knowledge areas: -	Principles: 2, 4
A review is a technique for systematically assessing the quality of a product. As a condition for a review, a separate description should be made of the requirements that the product needs to meet. The requirements may be described by a list of contents (if the product is a report) or by a set of criteria that the product must meet. The actual review consists of a meeting that lasts for a few hours. The reviewers' task is to point out what is good about the product, as well as its possible errors or shortcomings.	

Observations and story card summaries	
Knowledge areas: D, E, F	Principles: 1, 3

Overall, observation provides firsthand experience of work practices. In participatory observation, the observer is drawn into the performance of the observed work. Observation is time-consuming and usually requires advance identification of relevant work domains for observation. Project group members are recommended to set down insights from their observations in summaries, which are divided among the project group as starting points for further analysis. (p. 210). In the ERP design project, the participatory observations sessions are summarized as representations for cooperative understanding through a practice based on story cards (Beck and Andres, 2004) where work situation descriptions (Kyng, 1995) are written. These are intended to remind (not define) the design team of work situations in particular need of computer support.

IT-professionals overtaking work functions	
Knowledge areas: D	Principle: 3
Given the challenges faced in the project, in addition to the PD tools and techniques put forward by Bødker et al (2004), IT-professionals temporarily overtaking work functions were used as one additional PD tool and technique,	

Workshops	
Knowledge areas: A, B	Principles: 1, 2, 3, 4
To allow for genuine user participation, workshops involve users and IT designers working together to produce a joint result according to a relatively focused theme. A workshop can be organized in a variety of ways. They all use simple diagrams or drawings to gather and analyze the participants' knowledge about the theme. A workshop may be mainly analysis-oriented, for instance, aimed at a common understanding of a specific aspect of the users' present work practices. It may also be mainly design-oriented, and aimed at creating relations between the functionality of a series of new IT systems or between the systems and a proposal for a new work organization (coherent visions). Simple diagrams or drawings made in the process may appear later (p. 211).	

Experimenting with prototypes

Knowledge areas: B, E	Principles: 1, 2, 4
Experimenting with prototypes supports genuine user participation and it helps anchoring visions. It is a technique for generating ideas for IT usage and making simple prototypes for testing and producing concrete experience relevant to the further design efforts. Even a simple prototype or mockup, quickly and inexpensively put together, can have specific and tangible qualities resembling the envisioned IT usage. Hence, knowledge from testing and assessing prototypes can be put to constructive use in the further design efforts. It can be distinguished between experimenting with horizontal and vertical prototypes. A horizontal prototype has no functionality and can only be tested in an artificial setting. A vertical prototype is more complete (within a defined area of the design) and can be tested in real work situations to yield valuable experience about the expediency of its construction, content, and usage contexts (p. 215). Experimenting with prototypes consists of three activities: developing the prototype, testing the prototype, and evaluating the experiences (p. 293).	

Developing scenarios	
Knowledge areas: B	Principles: 1, 4
Developing scenarios is a technique that supports building coherent visions, and this helps in anchoring these visions. Scenarios visualize the practical application of a proposed IT system – that is, the potential effects of implementing it. Scenarios are prose-style representations exemplifying a work practice under future use of the system. Scenarios may illustrate application of the system as viewed from the different users' perspectives. (p. 216). The purpose of a scenario is to illustrate relationships between IT systems, work organization, and qualification requirements in a way that enables management and staff to assess whether their goals and needs have been met.	

Excerpt 8.1 The PD knowledge framework and Tools and Techniques used

8.2 CMD 3 phase 2: Application of tools and techniques used in the design study described

This section gives an account of the empirical material used in the practical course of application of the PD tools and techniques in the ERP design project. How their application worked to specifically support shop floor domain experts to talk about technical and organizational infrastructure matters is evaluated in the next section. This is primarily a description of how the tools and techniques have been used, and the order should therefore only be seen as a schematic indication to an in-practice evolving and iterating process.

8.2.1 High-level requirement soliciting from the different departments

Functional analysis

Version 100905

Technical

- Database:
- End-User configurable processes:
- Modular Architecture:
- Workflow editor:
- Web-based client:
- Workflow engine:

Breakdown of functionality for identified business areas

- Finance / Accounting
  - General Ledger (treasury)
    - Accounts Payable:
    - Accounts Receivable:
  - Allocations:
  - Multi-companies:
  - Responsibility Centers:
- Human Capital Management
  - Benefits Administration:
  - Compensation / Expenses Management:
  - Employment Contract:
- Project Management
  - Operational Management
    - Task allocation:
  - Financial Management
    - Costs:
    - Budgets:
  - Timesheets:
- Reports / Analytical accounting:
- Business Process Management:

Figure 8.2 Functional categories breakdown

During an early stage in the project, a high-level functional breakdown was solicited from the department managing the four functional areas targeted in the project. Based on internal document analysis (intranet, manuals, and written documentation) from the different departments and external information gathered from the Internet, this researcher started to compile an overview with general functions of relevance for different departmental areas. The functional overview was not tied to the technical functionality of any particular provider; rather, it was used as an early input to people on the shop floor of relevant functions they both already were working with and sought to acquire. The functional breakdown was then complied through an iterative process by mail correspondence and early introductory meeting between both departmental managers and administrators / assistants and this researcher. Figure 8.2 shows an extract of the compiled functional categories document. On average, 20 high-level functional requirements were identified for each department.

8.2.2 Contact with reference companies

Company visits

Reference companies were contacted both early and late in the ERP project to get to know about their ERP implementation experiences. Early in the project, this researcher and the other internal IT-professionals visited nearby universities to understand how they had approached their ERP implementation project. In addition, two close affiliates to the university were contacted: BIMCO, a Danish shipping association that was represented in the university's governance structure; and IMO, which is WMU's parent organization within the United Nations structure.

At the end of the project, a number of potential vendors' reference customers were contacted through Skype and domain experts were selected to follow-up on design assumptions made in the story cards of pieces of functionality in the system being designed (8.3.6).

8.2.3 Identifying providers

Review

Based on the breakdown of the functional categories and recommendations and non-recommendations made by the reference companies, a number of ERP providers were identified and contacted. In the contacts with potential vendors, the high-level functional requirements were used as check lists to understand what functionality they could provide in two ways: (1) The vendors were sent the lists and asked to indicate which functionality they could provide; (2) this researcher went through information material provided and acquired through the internet – with input from domain experts. This was an iterative process where the list of functional requirements evolved further. In the end, four ERP providers were selected for further evaluation by the IT-steering committee and IT-group: Visma, Microsoft Dynamics, OpenERP, and Agresso.

8.2.4 Participatory observations with departmental functions resulting in story cards

Observations

For each of the departmental functions, critical work processes were identified by shop floor people, the IT-steering committee, and the IT group, and based on these, participatory observation sessions were carried out. Each participatory observation session would typically last a few hours. A shop floor person identified as a domain expert of a particular work process and this researcher agreed in advance on a meeting time and what work processes would be targeted. Then we would sit down in the domain expert's office and would together work through the work processes agreed on. For the finance and project management functions, two participatory observations were carried out with Finance Officer Ocampo and Senior Finance Assistant Garcia. For the Human Resources functions, two participatory observation sessions were carried out together with Human Resource Officer Olsson. The registry function was comparably the most comprehensive functional area. Three participatory observation sessions were carried out with Registrar Davis, and two participatory observation sessions were carried out with Registry Assistant Magnusson.

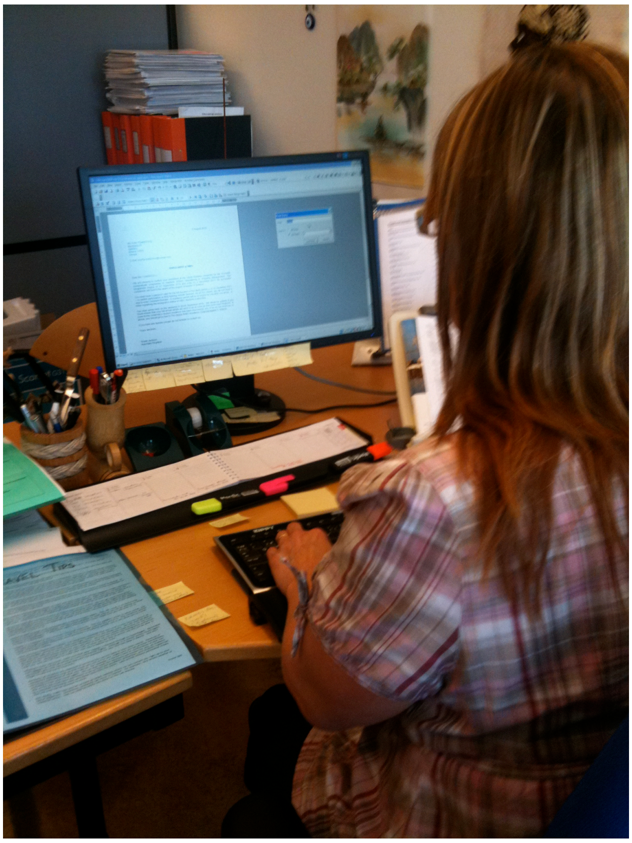


Figure 8.3 Senior Registry Assistant Magnusson at desk during participatory observation

To summarize the participatory observations, a first story card version was written by this researcher based on his impressions and then iterated back and forth with the respective domain experts until a final version was agreed on. As Figure 8.4 exemplifies, they were written from the perspective of the domain experts and their workflows, deliberately referencing their domain specific terminology (in the story card, the term "skeleton" were, for example, used to denote how a student profile was initiated in the current registry system). The story cards also referenced departmental specific systems, reports, documentation etc. Upon completion, the story cards were publically shared with all internal stakeholders of the ERP project.

Story Card: Grade Entry

Current workflow description of the Student Records Database:

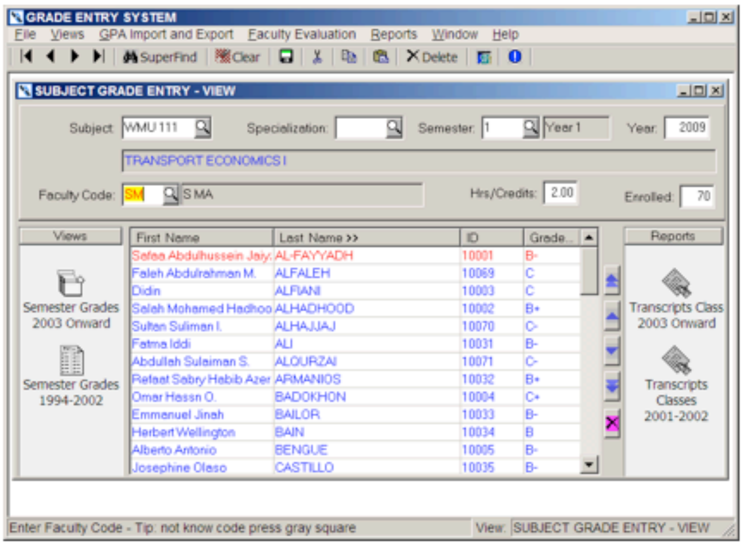


Figure 3. Subject Grade Entry interface

"Grade entry" and associated reports used to review and analyze the grades are the main functions of the Subject Grade Entry part of the Registry system.

- There is a workflow chain staring with a short definition "skeleton" found in the academic handbook which defines the overall structure of each of the specializations and provides a list of the subjects offered, in addition the course scheme documentation or the "meat" provides a detailed outline of the subject content and assessment for the subject, and finally the grades are entered by the Registry staff into the grade database once the assessments have been completed for each of the subjects taught. (see Academic Handbook and Course Scheme Documentation).
- The grade entry workflow starts when Susanna receives an e-mail notification from the course administration of WMU's intranet informing her that grades for a subject have been "approved by CAC and released to the students enrolled in the subject. In the "subject grade entry input screen a subject header is then created for that subject using the subject code and title that is listed in the applicable Academic Handbook.
- The e-mail sent to Susanna contains a link to a grade sheet interface on the intranet course administration system, which lists the grades that all students received for each subject taught. The grade sheet is printed out and the grades are entered manually into the Registry grade database using the "subject grade entry" screens (see Intranet Grade Reports).

Figure 8.4 example of a story card produced in connection to the participatory observation sessions.

8.2.5 An IT-Professional taking over work functions in "other" shop floor development constituencies

IT-professionals overtaking work functions

As stated, finding ways to better integrate the academic functions of the registry and faculty department was a comprehensive and complex part of the ERP design project.

As an IT-professional, this researcher had previously worked primarily with the academic functions from the faculty’s perspective. Following the participatory observations in order to better understand certain workflows, he ended up temporarily taking over / assisting the registry department in certain aspects of their daily work for given periods of time, for example, assisting the registry staff in defining electronic reports against the registry database.

8.2.6 Cross-departmental rich picture workshops

Workshops (analysis and design oriented)

The story cards were used as input for rich picture workshops with cross-departmental shop floor people. In total, 10 rich pictures were created. Figure 8.5 shows an example of how the academic admission workflow between the registry and faculty department was mapped into a rich picture. As the rich pictures were not tied to a specific department, they were made available on “neutral ground” in this researcher’s office - located in the middle of the WMU organizational landscape – during the course of the project. The basic layout of the rich pictures was constructed based on the story cards and approved by the respective domain experts. To illustrate a certain workflow, they mapped entities such as stakeholders, screen dumps from current systems, reports, and documentation. The workflow was indicated with post it notes. Workshops were then subsequently arranged, where, for example, faculty and registry staff would look at a rich picture to understand the nature of a workflow and come up with ideas of how it could be improved. The rich pictures were correspondingly modified.

Figure 8.5 Example of a workflow mapped into a rich picture pinned up on the wall in my office

8.2.7 Workshops / presentations with vendors

Workshops, Company visit, Experimenting with prototypes, Developing scenarios

Each of the four vendors identified made at least two onsite presentations / workshops. The first presentation was of an introductory nature to present the company and introduce its ERP solution. The engagements with the vendors can be characterized more as workshops in that each focused on certain functional aspects of their particular system. Typically, the staff of a particular departmental function was invited and targeted at a given workshop - although they were generally open for all staff to attend. Before each presentation, the vendors were sent a purposefully edited version of a story card on which to base their presentations. To this end, the vendors were asked to use the story card to input sample data to illustrate how a certain workflow would work in their system. In addition, two of the vendors invited staff that had worked onsite at other universities implementing their solutions as reference customers. The domain experts were encouraged to ask specific questions about system functionality. The vendor that was chosen in the end made six such functional specific workshops.

For each vendor, an internal evaluation file was set up that was updated after each presentation / workshop (see Figure 8.6). What the evaluation file documented corresponds to the second function of a story card of representations of the system being designed. Again, the story cards were written with the intention of being understandable for all cross-departmental stakeholders. The first part contained a general characterization of the system together with meta-data, such as when and how representatives from the vendor had arranged presentations. The second part of the story cards documented how the vendor met the departmental functional breakdowns (8.3.1). The final part of the story cards contained short stories of how the system was expected to work in each departmental area.

Agresso

Overview: Agresso contains modules for course administration, basic CRM, finance, HR, and project management. The focus of the system is to provide a complete and integrated solution that satisfies both financial and resource-related processes as well as the requirements and needs of information flows in an organization. Agresso is the most comprehensive commercially implemented ERP system evaluated in terms of functional modules and their scope. It is also a solution that is well established on the international market and in higher learning education settings: Harvard Business School, among else, uses the course administration modules of Agresso. In Sweden, Agresso is used by for example the Blekinge Institute of Technology (BTH) and in many public administration offices.
First pitch at WMU: 19th of August, Björn-Olof Bjälkander, Cecilia Baumann, Anne Jäglund, and Kristin Magnusson, Unit4 Agresso, Malmö.
Presentation of Education Management System (EMS) and Executive Presentation: 29 of August, Björn-Olof Bjälkander and Urban X, Unit4 Agresso, Malmö. Jacques and Yann, Unit4 Agresso, Paris

- Database: **MS SQL, Oracle**
- End-User configurable processes: **Yes**
- Integrates with Industry Standards: **Yes**
- Mail integration: **Group Wise, Lotus Notes, Outlook**
- Modular Architecture:
- Office Integration: **Microsoft Office**[1]
- Programming Language: **C# and C++ (.Net Framework)**
- Personal Dashboards: **Yes**
- Role-Based User Interfaces: **Yes**
- Supports SOA (service oriented architecture): **Yes (web services)**
- Supports Multiple Languages: **Yes**
- Supports Multiple Currencies: **Yes**
- User Portals: **Yes**
- Workflow editor: **Yes**
- Web-based client: **Yes**[2]
- Windows client: **Yes**[3]
- Workflow engine: **Yes**

Breakdown of functionality for identified business areas

- Customer Relationship Management: **Yes**[4]
  - Contact Management: **Yes**[5]

[1] Agresso Excelerator
[2] Agresso Self Service Application, purpose
[3] Agresso Smart Client, purpose
[4] The Education Management System (EMS) contains CRM functionality and can be extended with flexible fields and tabs. Otherwise, Agresso CRM plugin is available for example for Microsoft CRM (recommended).

Figure 8.6 Vendor evaluation file

8.2.8 The end, choosing a vendor

One after one, the vendors were counted out, until two final candidates remained. For each of these two vendors, strategic business cases were written and they were asked to submit contract proposals detailing the cost and timeline for the next phase of the design project and a later implementation project (also detailing training and maintenance, etc.). At the beginning of 2012, one vendor was finally chosen to provide the system based on the design study.

8.3 CMD 3 phase 3: Evaluation of how the tools and techniques supported domain experts in the design project

This section evaluates the application of the PD tools and techniques to support, in particular, technical and organizational infrastructure design dimensions.

In addition to the commonly documented function of PD tools and techniques, new dimensions of added value can be derived in the evaluation of the empirical material of the ERP design study. Through concrete examples from the empirical material, it is evaluated how heterogeneous users groups and IT-professionals learn from each other and how existing local development constituencies are re-designed. In addition to research diary notes and project documentation, the evaluation was supported primarily from the second timeline workshops with Evans and Magnusson (see chapter 4 for an overview). The evaluation is made from the perspective of the PD tools and techniques being applied through action research in a real project setting. The focus is thus not on the general traits of the PD tools and techniques, but on their function when being faced with real project constraints.

8.3.1 Functional analysis to support integrated project scoping

The functional analysis (8.3.1) made a necessary contribution to integrated project scoping in a PD way. Different sub-systems of the comprehensive ERP design project needed not only different amounts, but also different types, of attention. When PD is used beyond the local project, this becomes an important consideration when faced with project constraints, such as time and access to people and resources, in an organizational setting.

The functional analysis was thus not applied in the comprehensive rationalistic and expert-oriented way documented by Bødker et al. (2004, p. 235; 2004), which primarily supports managers to map and change functional entities. In this case, the functional analysis instead became useful in a more agile manner to solicit high-level requirements together with people on the shop floor:

- On the one hand, the finance department could pin point 20+ functional requirements that could directly be related to the functional list templates provided by this researcher and the checklists returned from the providers. From previous employment, the finance officer, Ocampo, Head of Administration Anders, and Senior Faculty Assistant, Garcia, all had experience with finance systems as part of ERP solutions.

- Even though the registry had comparably the most comprehensive technically integrated system support in place, it could not easily be mapped to standard categories. Registrar Davis stated in regard to the tailored registry system already in place, “What we do is we built the system to basically do our jobs, all of our jobs, all the four people in the registry, and that was exactly the purpose, to go from a manual paper based operation, to a computerized electronic method.” This implicated that the registry system was not necessarily following standard ERP templates of how to do registry work. (see chapter 5 and 6 for a more detailed description of the registry system)
- Finally, the Human Resource Officer, Olsson, who (alone) managed human resource business described how she had difficulties to immediately recognize what a new system explicitly could do for her.

The high-level requirements soliciting, through the functional analysis, shows how it is possible to negotiate the project scope together with the domain experts in an integrated setting. The examples together illustrate how it is possible to deal with the different parts of the system that need not only different amounts, but different types, of attention. This, for example, had implications for how the project came to work with story cards and participatory observations in order to understand the comprehensive, but customized, functions of the registry department. The high-level requirement solicited through the functional analysis thus illustrates how it is possible to negotiate the project scope together with the domain experts’ involvement.

8.3.2 Participatory observations and story cards to build mutual understanding in the re-design of shop floor design constituencies

The participatory observations and the iterative process of writing story cards supported the design of new design constituencies still anchored on the shop floor, thus extending the documented function of understanding functional pieces and workflows in a situated way, together with users. The participatory observations and story cards show how PD tools and techniques can be used to re-negotiate a PD design approach in an integrated technical and organizational infrastructure setting.

This dimension was emphasized in the follow up timeline interview with Magnusson and Evans. As accounted for by Magnusson in the pre-considerations leading up the design study, herself and the registry staff had enjoyed a close knit local shop floor design collaboration of the existing registry system, but now this was about to change. Evans noted that there was a certain apprehension and distrust in the beginning about leaving the "symbiotic relationship" that had existed before when Registrar Davis had also technically developed the registry system himself in close collaboration with his colleagues: "Davis knew exactly what we were doing everyday [...] our tasks were not glued up with the rest of the building and it was just a very immediate close work relationship." The new ERP design constituency for the registry would entail myself working as an intermediate between the ERP vendor and the registry staff to configure the system and assist in setting up reports. In this sense, the participatory observation, the iterative process of writing the story card, and this researcher temporarily taking over some registry work functions worked as a vehicle to build up domain specific understanding of the registry department. As Evans noted, "There has to inevitable [sic], there has to be a trust between us the users and the [IT-Professionals]."

In the case of the registry system, there already existed full formal documentation about its function. However, just reading this documentation would arguably not have served the same purpose as the participatory observations and the iterative writing of the story cards. What is shown through the WMU example is how a PD technique can support a learning process where shop floor people gain confidence of a re-negotiated design constituency.

8.3.3 Rich-pictures to deliberate new design between shop floor development constituencies

The rich picture workshops are an example of how a concrete boundary object can function to build an integrated understanding of both existing work situations and deliberating on a new design. In the ERP design study, it was a challenge to integrate different local design constituencies with different interests, but ones that were working with overlapping functions and materials. In the functional area of academics, the defining of subjects and marking had been done completely separately between the registry and faculty and then manually transferred between the departments. To this end, the rich picture technique showed how heterogeneous shop floor people and IT professionals - coming into the design situation with different perspectives - could come together in an integrated setting to work out a new integrated workflow.

8.3.4 Story cards to enable critical evaluation of different technical infrastructure options by shop floor users

In a design study of a comprehensive technical and organizational infrastructure such as an ERP system, it is not feasible for vendors to showcase, or for shop floor people to evaluate all pieces of functionality. A central evaluation topic in the empirical material of the vendors' presentations and workshops was the extent to which the users felt that they could constructively and critically reflect on design ideas based on the ERP system functionality showcased - given their individual work domains - or if it was just marketing of the vendors systems.

As described earlier, both Registry Assistant Magnusson and Student Social Officer were a tough team, as they were in many respects already satisfied with the registry shop floor development constituency. During the interview, for example, they stated how they had issues with other vendors' presentations and workshops, where they did not get meaningful answers from the provider:

Evans: For me, there was a big difference, because [the other system presentation] hasn't been as targeted to our specific tasks; the people doing the presentations - it has very much been more marketing, in my opinion anyway.
Magnusson: just assuring us that it will…
Evans: yeah
Magnusson: …that it will work.
Evans: Any questions that you threw at them […] it just seemed like any question we posed or [the IT-professional] posed, oh yes, no problem -
Magnusson: Everything can be customized?!
Bolmsten: They want to sell their product?
Evans: Exactly.
Evans: Well, that's what you keep suspecting that they are saying yes, yes everything will be fine.

However, they perceived how there had been difference in the ERP design project:

Evans: It was a relevant presentation, and it addressed tasks that I was very much involve with, for example quality assurance, and doing student evaluations and so on.

Magnusson:Mm, and I remember like, wow, could you do that, and I was thinking already, how will I work with that, and got a little exited [laughter] -

Evans: and I think because of the groundwork you had laid, they were able to focus on relevant tasks.

The groundwork here was that the vendors were asked to base their presentations on the story cards sent to them, which were written based on a long-term engagement with the people in the different shop floor development constituencies. The vendors were also asked to input data from the story cards in their technical systems to illustrate specific pieces of functionality, to which several vendors commented that this was a novel way of working. They were used to first signing a contract and then initiating a feasibility study using their own project model. But, as Evans commented:

Evans: But, I mean, if you are buying a car you are trying several different cars, and you test drive them -
Magnusson: compare it.
Evans: You don't sign a contract with a car, and then test drive it.
Bolmsten*: How can you test-drive an ERP system, well, that's an interesting…?*
Evans: well of-course you can't, but you can at least see it in action or have the confidence increased that it is going to drive in the speed you like or turn left when you want it to turn left.

The PD approach taken in the ERP design project, in this way, shows how people on the shop floor can "test drive" also complex technical infrastructure to critically evaluate its usefulness.

8.3.5 Reference company contacts to test design assumptions

The contacts that were made both in the beginning and the end of the ERP design study with the reference customer, however, enabled rapidly testing critical design assumptions that were difficult to evaluate on an individual basis. At the beginning of the project, the management committee of the university put forward the ERP systems of the university's close affiliate of IMO as the primary option to consider. However, after contacting IMO and learning about their experiences, it was swiftly taken off the table. The second purpose of the contact with reference customers became to test the design assumption as an outcome of the story card reflections after the vendor workshops with other universities that had implemented the vendors systems.

Optimally, in accordance to the documented function of company visits, the stakeholders would have been invited to visit reference companies. This was also the initial plan. However, given the number of user groups involved and being geographically dispersed in France and the US, this was not feasible. In this sense, the reference customer contacts were the least participatory tool and technique used in the sense of directly involving domain experts. They instead served as a secondary “back-office” way to triangulate the internal design assumption made by domain experts and IT-professionals.

8.4 Summary

This chapter has shown how common PD tools and techniques can be configured to understand and deliberate a new integrated technical and organizational infrastructure. The need of working with representational means to support infrastructure deliberations was visible in all CMD research cycles in this PhD research. However, in this chapter this issue has in-itself been the focus of the deliberations and reporting.

Based on Bødker et al.’s (2004) knowledge framework – developed as part of the MUST method - a number of PD tools and techniques were selected and configured in relation to each other: functional analysis, company visits, review, observations and story card summaries, IT-Professionals overtaking work functions, workshops, experimenting with prototypes, and developing scenarios.

These were then empirically evaluated, based on specific dimensions that pertain to integrated technical and organizational infrastructure development – thus extending their commonly documented and applied value in a local project context. In this sense, the results make an incremental contribution to the MUST method when using PD in an integrated setting with heterogeneous user groups in regard to: how much focus has to be placed on existing subsystems was deliberated through the high level requirement soliciting; who the advocate is and why integration is important for different user groups were understood and deliberated through the story cards and rich picture workshops; and what the implications are that bring a search for an integrated solution with regard to the whole way of proceeding was understood and deliberated through the critical vendor evaluation.

9. Organizational IT management from the shop floor

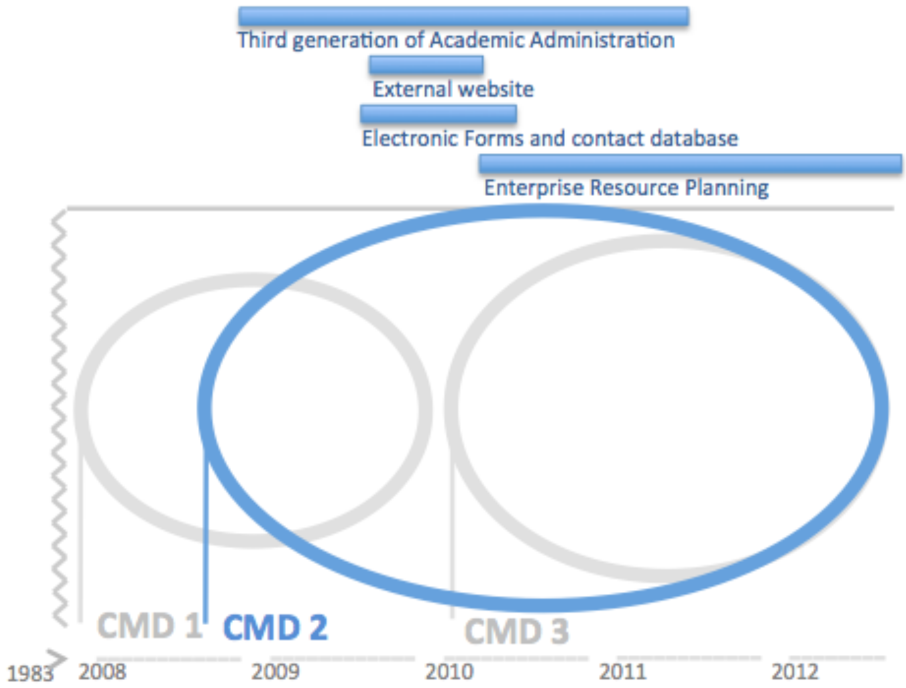


Figure 9.1 The second initiated Cooperative Method Development action research cycle of how people in shop floor development constituencies together with IT-professionals and managers can organizationally manage infrastructure development. The primary case used in the deliberations is an extended development of the academic administration software support complemented by EUD developments in the registry and administration shop floor development constituencies about an external website, electronic forms, and a contact database. In addition, the Enterprise Resource Planning design project reported in the previous chapter was part of the evaluation phase WMU (see complete timeline in chapter 3).

Users on the shop floor need to organizationally manage also the infrastructure that surrounds their local development of software support. This chapter reports on the second initiated Cooperative Method Development (CMD) action research cycle. The organizational IT management dimension of infrastructuring (Karasti & Syrjänen, 2004; Karasti & Baker, 2004; Pipek &Wolf, 2009) is in focus, showing how it is possible to leverage the capabilities of shop floor IT development on the organizational IT management arena. The two types of established shop floor IT management at WMU – with and without the support of IT-professionals, as presented in previous chapters - are joined and form a base for the action research deliberations of supporting the development of an evolving organizational IT management.

As is indicated in Figure 9.1 and as was described in the breakdown of the CMD application in chapter 4.3, the second action research cycle was the most comprehensive. The need to improve organizational IT management was prompted in conjunction with the technical base action research reported on in chapter 7. In addition, part of the ERP design project reported on in chapter 8 was included to evaluate the outcome of the improvements made. Similar to the development of an integrated technical base and Participatory Design (PD) tools and techniques for infrastructure development, improved IT management on the organizational arena was something that was called for by the users themselves. If coordination between isolated projects could be achieved, there were potential synergies to having a shared technical and organizational infrastructure. The empirical material shows how end-users could participate in the co-evolution of a decentralized organizational IT management, and how cooperative organizational decision structures and processes could be integrated with a shop floor IT management model based on PD and End-User Development (EUD).

A challenge from an organizational point of view, but opportunity for research was that useful guidance offered in related work was limited. In line with Ciborra's (2000) review "from control to drift" described in the related work chapter, the practices to be supported at WMU were from the outset found to “diverge from the wisdom contained in the management and IS literature of today.” However, the users were not satisfied with a "drifting" management of infrastructure either. The challenge of PD in the action research was involving users in the meta-design of structures and processes.

The following section (re-) introduces the shop floor development constituencies from the previous chapters that have been used to understand situated practice in its historical context. As an additional base for the action research deliberations, a failed attempt to implement a top-down organizational IT management is reported. The deliberations of an organizational IT management anchored with the shop floor are then described, followed by an account of key-dimensions from the evaluation of the empirical material collected.

9.1 CMD 2 phase 1: A need for an extended shop floor IT management in the organization

The two established shop floor development approaches at WMU introduced in chapters 6 and chapter 7 provide the situational context within which the action research set out to deliberate and to improve organizational IT management. As re-iterated below, in both cases, there was a desire to continue to cultivate the existing shop floor IT management approaches:

- **Shop floor IT management with IT-professionals:** The academic administration shop floor development constituency was the first that this researcher encountered and became part of when employed as an IT-professional at WMU, taking over the development of the academic administration software support in 2005. This later became a core project in the action research deliberations, accounted for in section 9.2. As described through the account of Vice-President Wang in chapter 7, there was a perceived benefit in having end-users and IT-professionals working close together, and also sharing work assignments and even offices in the development of software support. A long-term ambition of the development of the technical base of the academic administration was to both improve "use in design" in the daily collaboration between IT-professionals and shop floor users and further enable "design in use" for faculty assistants to carry out development and configurations themselves.

  Shop floor development without IT-professionals: Apart from collaborating with IT-professionals, members of WMU developed their own tools. As was alluded to in chapter 5, in a number of cases, end-users even took on technical infrastructure development themselves without any primary dependence on IT-professionals. Starting in 1992, Registrar Davis gradually began to construct an accountable registry management system that included computerized support together with his staff. The rationale for this approach to software development was to get useful software support. In a similar manner, Administrative Assistant White continued to manage the technical development of electronic forms and contact database, describing herself as "sort of a spider in the net," both in regard to working with users and IT-professionals.

These two complementary accounts of how development of software support practically taking place by users themselves give an indication what the organizational IT management had to cope with.

9.1.1 Making improvements to established structures

The motivation to improve the organizational IT management came with a need for better integrated technical and organizational infrastructure. While people working on the shop floor often were satisfied with the existing local development practices, they also recognized the opportunities (as well as challenges) of synergies in terms of improved quality and efficiency of different local software systems being integrated. In reference to the examples in the previous sections, a lot of cumbersome manual work had to be carried out to transfer marks between the faculty and registry system, which not only took up time but also caused mistakes. For this purpose, a number of organizational IT management structures already existed that later came to underpin the action research deliberations:

- The IT coordination group: Connected to the first CMD research cycle about the technical base accounted for in chapter 7, an IT-coordination group was initiated. The three IT-professionals at WMU were formally sorted under the management of different individual departments. To this end, the IT-coordination group provided an IT-professional-forum for coordination chaired by the "Head of Information."
- The IT-steering committee: The IT-steering committee gathered user, IT-professional, and management representatives from the different shop floor development constituencies. As described in chapter 5 in regard to the organizational IT management that existed already before this research study, this was one of the earliest committees at WMU with a history dating back to 1983. In realizing a need for organizational IT management, the committee was initiated by user and domain expert representatives from the shop floor development constituencies themselves. The development of software support, such as the academic administration, the registry system, and the electronic forms and contact database, were reported to the IT-steering committee. Representatives from the shop floor development constituencies and IT-professionals at WMU served as committee members in accordance to a rotating scheme.

- The management committee: The management committee, chaired by the president of the university, brought together heads of the different departments such as administration, faculty, finance, and registry. The IT-steering committee's formal function was to act as a sub-committee to the management committee, providing advice on hardware and software capacity, maintenance, as well as advising on present and future demands for information technology provision and its co-ordination within the university. In practice, the IT-steering committee operated with relative freedom and also managed the annual computer budget.

In their functioning at the time, these organizational IT management structures were not suited to take on the increasing coordination needs for technical and organizational infrastructure. As the Head of Information reflected in the following regarding the challenges of integrating the development going on in the different shop floor development constituencies: "IT R&D priorities are intimately linked to organizational requirements. Strong personalities can and do skew the ranking of such priorities as everyone is number one in line when it comes to an expressed need." The need to improve organizational IT management was recognized already in the first CMD action research cycle (chapter 7). The third reflection paper used a concrete example of academic administration to illustrate that changes to the technical infrastructure were insufficient, also using the current shop floor IT management approach as a base to illustrate how much better management needed to be developed in both the local project and organizational arena. This later came to prompt the action research reported here.

9.1.2 A (failed) top-down organizational IT management

The first organizational response to the above challenges was to hire an outside IT-manager to implement a "best-practice" type of IS management. The account of these efforts contextualizes the following action research deliberations reported in section 9.2.

Before the action research deliberations, an attempt was made to put a traditional IS management in place at WMU. An "Information Services Strategic" (IIS) plan was defined that detailed a standardized and comprehensive IS management scheme. One of its main paragraphs was that the management was to define a number of "realistic, obtainable, sustainable, and measurable" service level agreements that would define the organizational IT management. These would then be implemented across the local shop floor development constituencies through a set of pre-defined maturity stages.

This also strategically implicated a new take on the functioning of the organizational structures in place: (1) the shop floor IT management model would primarily be reduced to adaptation of implemented software, (2) the role of the IT-steering committee would be reduced to an advisory / announcement forum, (3) and the IT-professionals would formally exist under the IT-coordination group who would assume an extended organizational IT management role over software support development.

Once implemented, this was intended to result in local development requests being made to the IT-coordination group that would, in turn, be vetted and prioritized against service level agreements and pre-set quality goals defined by the management body, and then announced in the IT-steering committee. According to the plan, this would step-by-step enable an integrated infrastructure both at a technical and organizational level.

The core objectives and strategy of the ISS plan were established with the president to secure a change mandate. Following this, three months were spent on conducting interviews to identify, analyze, and concretize service level agreements of how existing work processes and IT systems on an operational level could achieve their strategic potential through the established framework.

The result, however, was not what was intended. The ISS plan was rejected by stakeholders in the shop floor development constituencies. Upon launch, in regard to a concrete development instance, the IIS plan proposed that on a technical infrastructure level all local databases would be exchanged to a common database format. To this end, users, domain-experts, and managers stepped up their critique, both inside and outside the IT-steering committee. Registrar Davis, for example, vocally complained that his local needs and those of other individual departments were not sufficiently understood and that the route for change defined could not cater to the future needs of the departments. It also resulted in friction between the representatives from the shop floor development constituencies to the point that the IT-steering committee was temporarily taken out of operation. This course of events contributed to the responsible IT-manager leaving WMU.

9.2 CMD 2 phase 2: Deliberating a shop floor organizational IT management

The failed nature of the IIS plan created a mandate for organizational IT management from the shop floor supported by action research. As a member in the shop floor development constituency managing the academic administration development put it: "*The message is not that the current situation can't be improved. We clearly need a better software organization. But let's implement one that fits with the activities and nature of the organization.*"

The deliberations were based on an evolutionary approach that intended to cultivate the intended function of the original structures and processes: the shop floor IT management approach and the empowered function of the IT-steering committee in relation to the management body, and the IT-coordination group as a supporting function. The deliberations entailed the following measures to further collaboration amongst an increasing number of participants in projects and between local projects and the coordination structures. The intention was to continuously make IT-professionals available as resources for local shop floor development constituencies, rather than IT-professionals steering local shop floor IT management:

- **Story cards to improve coordination in local development:** The practice of story cards was decided as a means of enable more shop floor users to participate in evolutionary development in the local development projects, as well as to create a foundation for a linkage between the local development and organizational IT management and infrastructure development. The original shop floor IT management model, described in section 9.1, did not necessarily depend on any formal project management framework or documentation of the development. The challenge that the story cards targeted was to support the original shop floor IT management model as much as possible, but enable more users to participate in project planning. The development of the story cards was based on Beck and Andre's (2004) planning game and Kyng's (1995) paper on making representations work. The intention was to remind (not define) participants of current work situations in need of change and to present a vision for overall change, entailing both new system design and work practice design. As Figure 9.2 illustrates, the first part of a story card gave an overview of a concrete area in need of change. The second part of the story card then presented a vision for overall change, both regarding technical functionality as well as user interfaces and work organization. The final part of the story card detailed a technical and organizational implementation. As is also visible in Figure 9.2, the story cards represent both the system being designed and the work situations. Story cards could be continuously added during the development process, and their priority were then ranked primarily between users and IT-developers, with input from the IT-steering committee. An extended use of representational means was another change meant to support the involvement of an increasing number of users in design and implementation projects. Representational means used in projects are described in the sections below and exemplified in Figure 9.6 and 9.7.

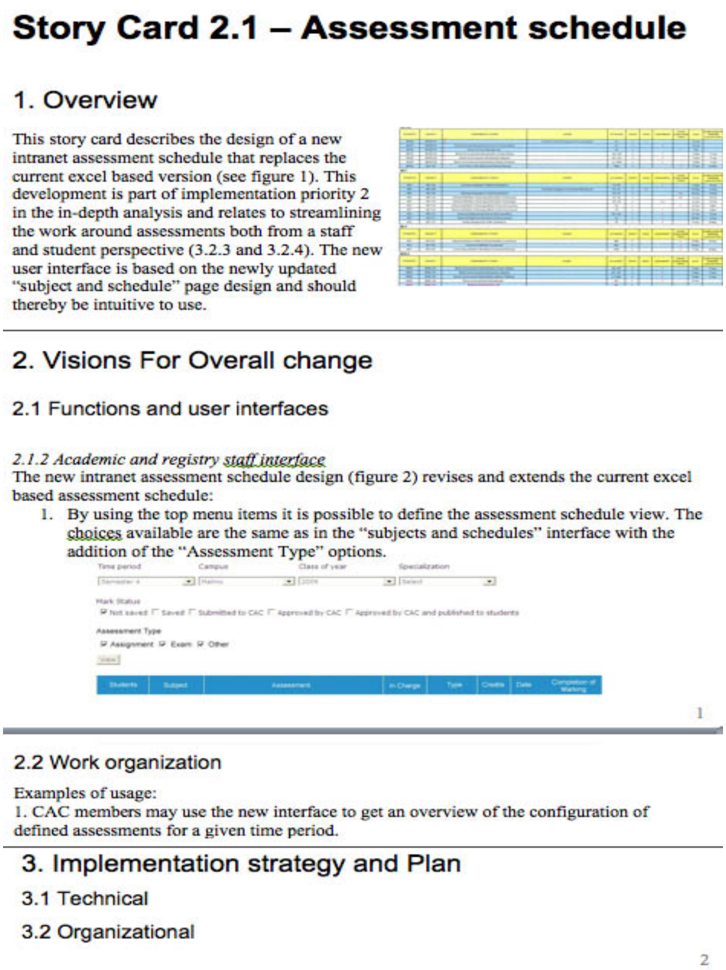
- **Participatory project management to expand shop floor IT management**: The second deliberation concerned the adaptation and use of the MUST method (Bødker et al., 2004) as a local project management framework. The MUST method was proposed due to its coherent user-centered design project management framework that builds on PD principles such as: a coherent vision for change between IT systems and work practices, genuine user participation in IT-design, IT-designers having firsthand experience with work practices, and anchoring visions for change with affected user constituencies. The MUST project management framework, however, needed to be customized to include not only the design, but also the implementation, project. As Figure 9.3 shows, the guidelines for developing a project charter was used to define the premise of a project, including assignment and objective, financial and technical framework, and key critical factors. The second phase of the MUST guidelines is an inline analysis to connect to strategic organizational IT management considerations. To support this function, the inline analysis was expanded to connect to the business plans described below, allowing the discussion of the relation and dependencies between different projects. The third phase of the MUST method guidelines concerns an in-depth analysis to understand the particulars of a work domain in need of change. In the adaption of the MUST method, the in-depth analysis was customized to describe current work procedures and problems, needs, and ideas for solutions to provide a context for the story cards. Finally, instead of ending with a design proposal (preceding a later implementation project) in the original MUST model, the last phase of MUST was extended to detail an evolutionary implementation process of the story cards. The idea from the start was not to implement a one size fits all version of the MUST method. As was emphasized by the chair of the IT-coordination group in already one of the first workshop meetings about how the MUST method could be used: "*MUST, I think the question is do we need streamline it, is it working now, is it too much work for some of the smaller projects or is there something that we can even scrunch and use for different size projects or more technically.*"

- **Business plans to coordinate organizational IT management**: To create an organizational IT management anchoring of the MUST project management framework and to link local development projects, a notion of 'business plans' was developed. The business plans were created annually and positioned in upcoming and ongoing development projects. They connected to the inline analysis of the MUST method and were used to create accountability of how the IT-professionals could be used as resources for development projects to the IT-steering committee and management committee. In their final version, an "action plan" was added to the business plans to report on how the development with the story card planning progressed during the projects.
- **Improved procedures in the IT-steering committee**: The final deliberation concerned improved procedures in the IT-steering committee. The procedures included: how agenda items were to be prepared in making use of the story cards, MUST project planning, and business plans; improved decision procedures to ensure that representatives from the shop floor development constituencies had a say in the matters discussed; and how minutes and follow up actions were to be defined.

The improved processes and procedures for organizational IT management planning were discussed and implemented in a bottom-up manner.

An early approach of boundary objects that was related to already as part of the first CMD research cycle was referred to as "reflection papers." These were intended to enable reflection on infrastructure matters between users at the shop floor, the IT-coordination group, the IT-steering committee, and the management committee. They described a concrete technical and organizational infrastructure issue from the perspective of a shop floor user. One reflection paper, for example, used a concrete example of academic administration to illustrate that changes to the technical infrastructure were not enough, and used the current shop floor IT management model as a base to illustrate how better management needed to be developed on both the local project and organizational arena.

Throughout the action research, the usefulness of the story cards and MUST project management framework was tried out in local shop floor projects. The progress would then be discussed in weekly IT-coordination group meetings. In addition, the IT coordination group carried out two workshops where the MUST framework was discussed, adapted and extended. On a continuous basis, the IT-coordination group would report the progress to the IT-steering committee where further deliberations would take place in refining the story cards, MUST project management framework, as well as the business plans and improved procedures. During the action research, the IT-steering committee became an increasingly important venue, compared to the IT-coordination group. The project development examples below illustrate the evolution and appropriation of the processes and procedures for planning .



Story Card 2.1 – Assessment schedule

1. Overview

This story card describes the design of a new intranet assessment schedule that replaces the current excel based version (see figure 1). This development is part of implementation priority 2 in the in-depth analysis and relates to streamlining the work around assessments both from a staff and student perspective (3.2.3 and 3.2.4). The new user interface is based on the newly updated "subject and schedule" page design and should thereby be intuitive to use.

2. Visions For Overall change

2.1 Functions and user interfaces

2.1.2 Academic and registry staff interface
The new intranet assessment schedule design (figure 2) revises and extends the current excel based assessment schedule:

1. By using the top menu items it is possible to define the assessment schedule view. The choices available are the same as in the "subjects and schedules" interface with the addition of the "Assessment Type" options.

Time period Campus Class of year Specialization
Mark Status
Not saved Saved Submitted to CAC Approved by CAC Approved by CAC and published to students
Assessment Type
Assignment Exam Other

1

2.2 Work organization

Examples of usage:
1. CAC members may use the new interface to get an overview of the configuration of defined assessments for a given time period.

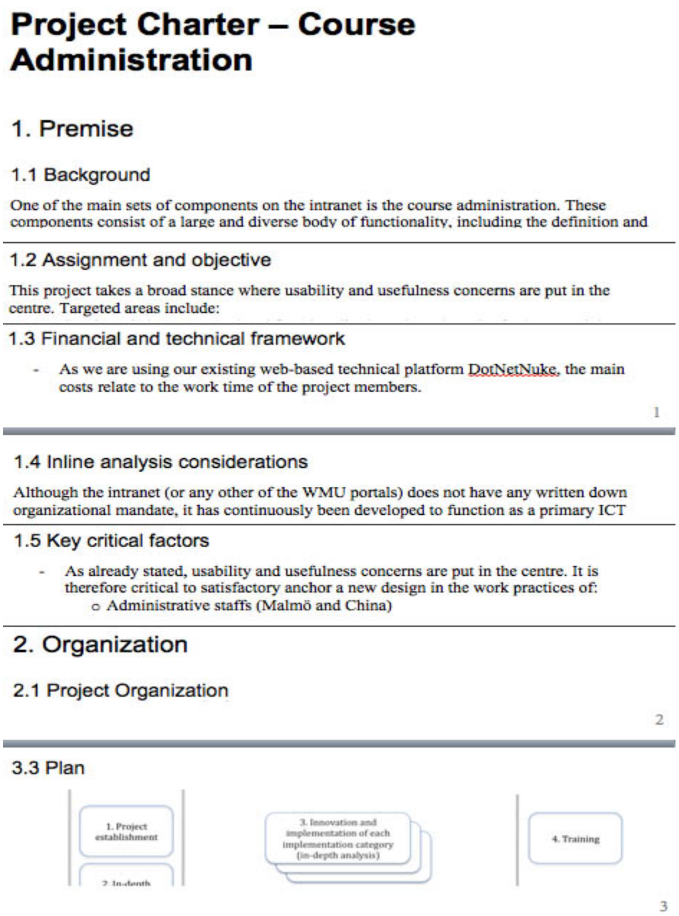
3. Implementation strategy and Plan

3.1 Technical

3.2 Organizational

2

Figure 9.2 Excerpt from story card about course administration



Project Charter – Course Administration

1. Premise

1.1 Background

One of the main sets of components on the intranet is the course administration. These components consist of a large and diverse body of functionality, including the definition and

1.2 Assignment and objective

This project takes a broad stance where usability and usefulness concerns are put in the centre. Targeted areas include:

1.3 Financial and technical framework

- As we are using our existing web-based technical platform DotNetNuke, the main costs relate to the work time of the project members.

1

1.4 Inline analysis considerations

Although the intranet (or any other of the WMU portals) does not have any written down organizational mandate, it has continuously been developed to function as a primary ICT

1.5 Key critical factors

- As already stated, usability and usefulness concerns are put in the centre. It is therefore critical to satisfactory anchor a new design in the work practices of:
  - Administrative staffs (Malmö and China)

2. Organization

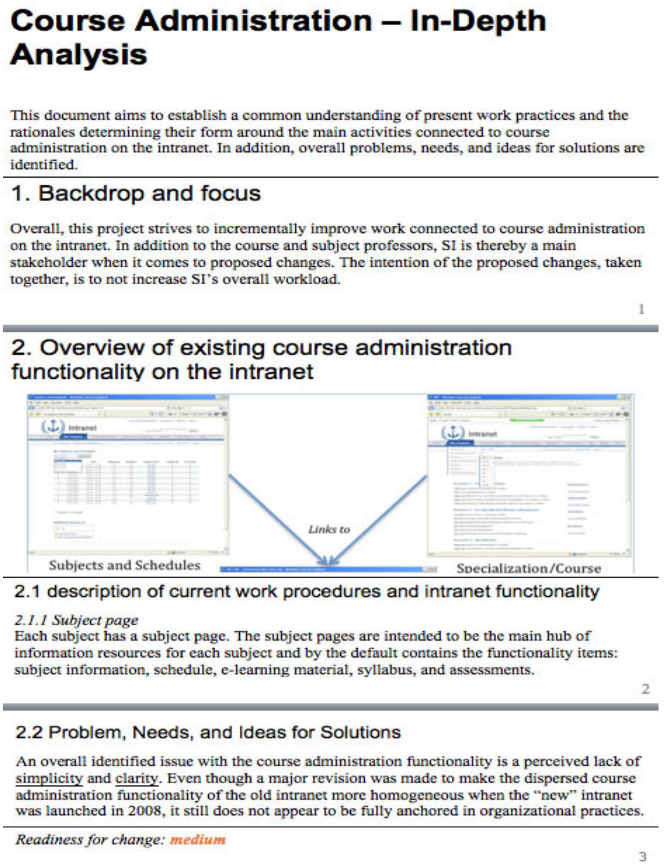
2.1 Project Organization

2

3.3 Plan

3

Figure 9.3 Excerpt from project charter about course administration



Course Administration – In-Depth Analysis

This document aims to establish a common understanding of present work practices and the rationales determining their form around the main activities connected to course administration on the intranet. In addition, overall problems, needs, and ideas for solutions are identified.

1. Backdrop and focus

Overall, this project strives to incrementally improve work connected to course administration on the intranet. In addition to the course and subject professors, SI is thereby a main stakeholder when it comes to proposed changes. The intention of the proposed changes, taken together, is to not increase SI's overall workload.

1

2. Overview of existing course administration functionality on the intranet

2.1 description of current work procedures and intranet functionality

2.1.1 Subject page

Each subject has a subject page. The subject pages are intended to be the main hub of information resources for each subject and by the default contains the functionality items: subject information, schedule, e-learning material, syllabus, and assessments.

2

2.2 Problem, Needs, and Ideas for Solutions

An overall identified issue with the course administration functionality is a perceived lack of simplicity and clarity. Even though a major revision was made to make the dispersed course administration functionality of the old intranet more homogeneous when the "new" intranet was launched in 2008, it still does not appear to be fully anchored in organizational practices.

Readiness for change: medium

3

Figure 9.4 Excerpt from In-depth analysis about course administration

9.2.1 Story cards and the MUST method in three projects

In a first workshop, this researcher presented the original MUST method (Bødker et al., 2004) and early ideas about how it could be appropriated together with the story cards. In addition, a first discussion of business plans proposed by the Head of Information took place. The presentation was later repeated for the IT-steering committee and the management committee. The outcome was a conceptual understanding of what using new planning methods would entail. The importance of not adding "bureaucratic red tape" (IT-Steering committee member) to shop floor IT management and implementation projects was highlighted.

External website

When the changes to the organizational IT management were discussed, an *external website* project had already been initiated. The project's strategic mandate had been decided, problems as well as potential solutions for a new design had been isolated, and the current technical solution and work processes had been identified. The old external website carried an outdated interface design and could only be updated by one person. The solution was a new design with a new technical platform and a distributed organizational work arrangement to update the website. It was, therefore, considered "red-tape" to halt the ongoing development process to conduct a new inline and in-depth analysis and write story cards. However, a first MUST project charter was defined to coordinate the ongoing development between users and IT-professionals. In addition, new PD tools and techniques as positioned by the MUST knowledge framework were used to involve more users in the design process. As Figure 9.5 illustrates, picture mock-ups were used to support the design between users bringing different ideas to the table. The left picture shows early design sketches by one of the IT designers and members of the original shop floor development constituency. The right picture shows how Administrative Assistant White – who had previously not been involved in the external website development – was able to contribute to the final design with ideas picked up from Stanford's external website.

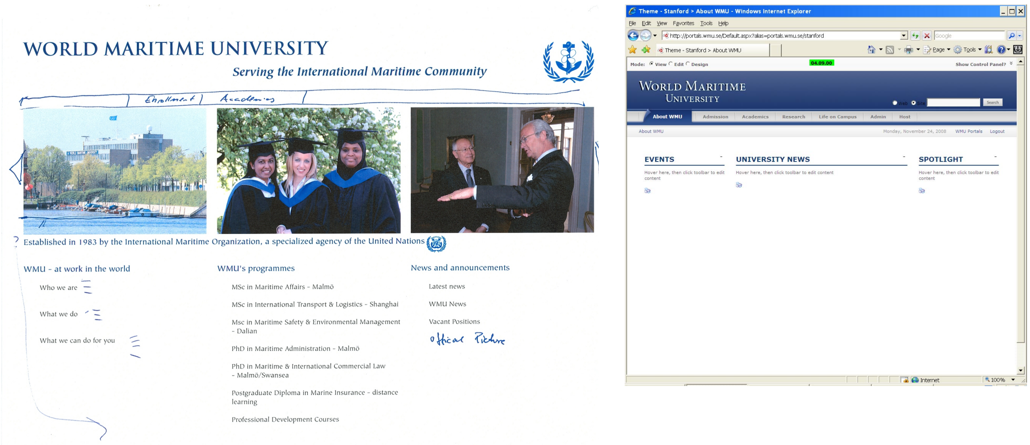


Figure 9.5 Picture mock-ups of external website design

Electronic forms

The second project also involved Administrative Assistant White and concerned the design of new electronic forms. As also presented in section 9.1, she had over a long period of time designed WMU's internal forms, for example, concerning leave of absence for both staff and students. For this purpose, she had used Microsoft word. The question now was whether she could design the forms using a new technical system that connected to a shared database developed by this researcher. In total, approximately 20 different forms existed. In this case, a project charter was developed, which also contained a strategic inline analysis that was vetted with the IT-steering committee. Following this, an in-depth understanding of White's current design procedures was sought through a number of participatory observation sessions. A report of the in-depth analysis was published to the IT-steering committee. The new design of the forms was based on a number of mock-ups, where design ideas were solicited from different stakeholders. Figure 9.6 shows an example of a design proposal with revision notes from a faculty professor.

Submit by Email | Print Form

WMU Proposal Budget

Project Title:

A. Salaries	Person-Months Project / WMU	Project	WMU Match
Principal Investigator/Project Director:			
1. Co-Principal Investigators:			
2.			
3.			
4.			
5.			
WMU Staff:			
Research Associates:			
Research Assistants / Ph.D Students:			
Total Salaries			
B. Benfits (including Provident Fund):		€0.00	€0.00
C. Total Salaries and Benefits:		€0.00	€0.00
D. Equipment Budget (items exceeding € 3,000)			
1.			
2.			
3.			
Total Equipment			
E. Travel (Destinations, Persons, Dates/Days):			
1.			
2.			
3.			
Total Travel			
F. Other Costs:			
1. Material and Supplies:			
2. Publication / Documentation / Dissemination:			
3. Computer / IT Services:			
4. Consultant Services / Subcontracts:			
Total Other Costs			
I. Total Direct Costs		€0.00	€0.00
~~J. Total Indirect Costs (40% of total direct costs):~~		€0.00	€0.00
K. Costs (I+J):		€0.00	€0.00

J. Indirect Cost (40% of Total Salaries and Benefits [C])

PI / PD Name and Signature

Figure 9.6. Electronic forms mock-up with revision notes

Course Administration

The most comprehensive example in utilizing the story cards and the MUST project management framework was a revision of academic administration functionality. The previous Figures 9.2, 9.3, and 9.4 show how the academic administration project made use of the MUST guidelines: project charter, inline analysis, and in-depth analysis. Seven main story cards were positioned that detailed a revision of subject, schedule, and assessment administration in the university's online academic portal. The understanding of current problems and ideas for solutions was developed through day-to-day interaction between users and IT-professionals. In addition, three workshops were conducted with different constellations of faculty (Figure 9.7 shows one of the rich pictures that were constructed during these workshops); student representatives were consulted; and starting ideas of faculty and registry integration were envisioned with the registry staff.

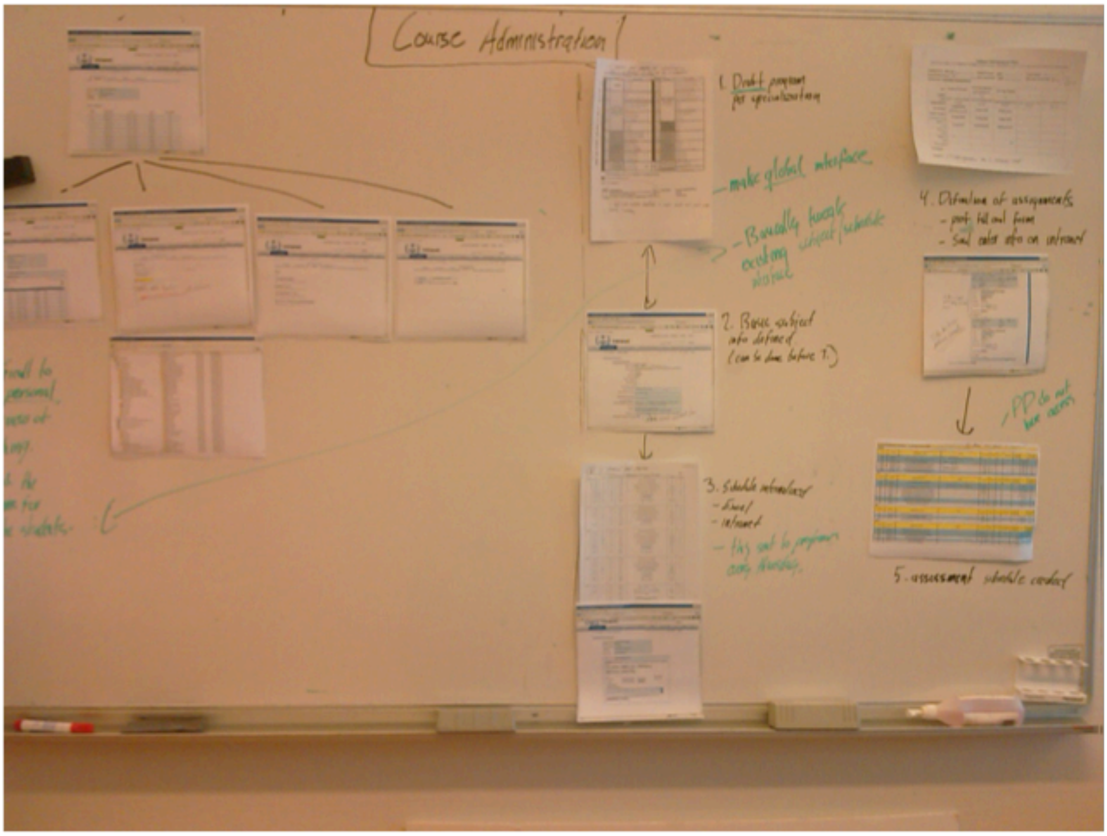


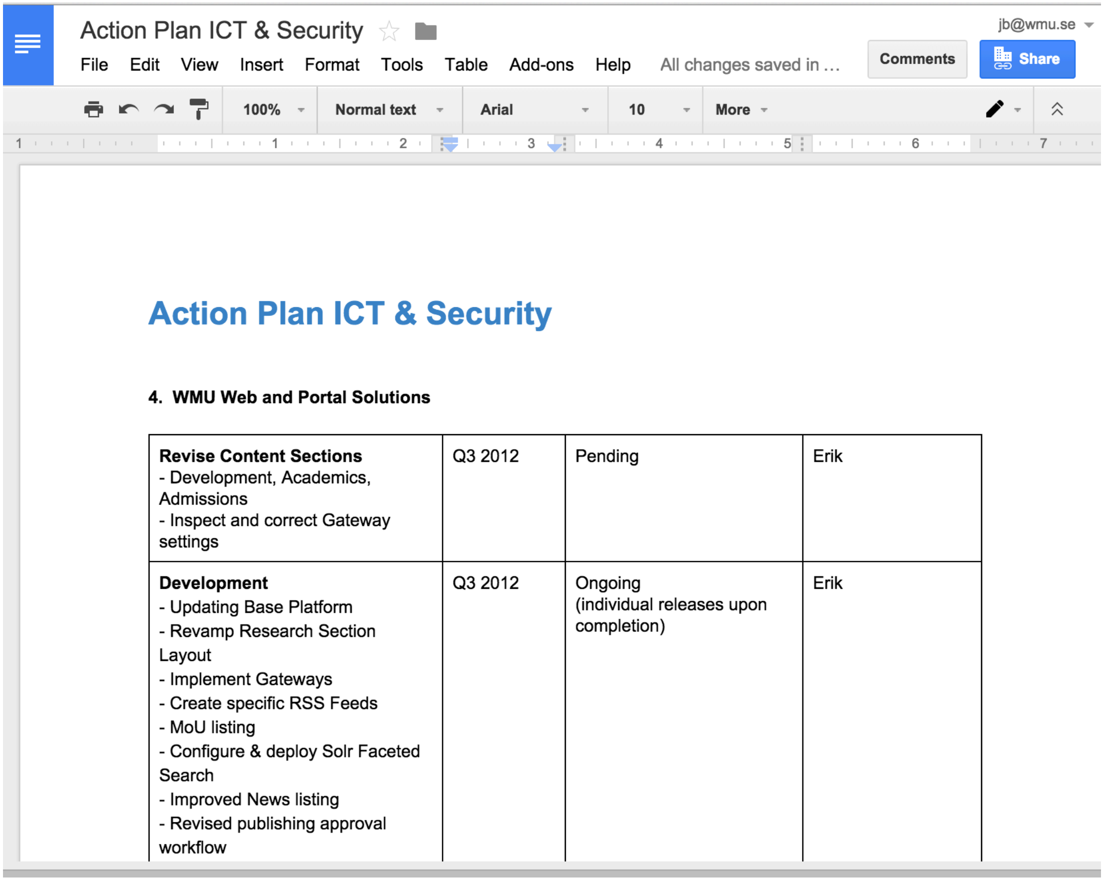
Figure 9.7 Academic administration rich picture

9.2.2 Business plans

Chair of the IT-coordination group Bouchard took the lead on developing annual business plans. The business plans complemented project planning through MUST and the story cards to support organizational IT management by the IT-steering committee and the IT-coordination group.

In this way, the development of integrated business plans supported also came to gradually support a function to coordinate the work between the IT-professionals that were formally existing under the management of different individual departments. The IT-coordination group thus formally had an organizational subordinate role, where local departmental shop floor development constituencies had precedence.

The business plan procedure was gradually evolved during the course of two years: a trial version of a business plan was first defined that compiled ongoing and upcoming project work, including the three projects accounted for above. This version was used to test the concept of using business plans for coordination between the IT-coordination group, the IT-steering committee, the management committee, and the management of the respective departments. In addition, as described above, some of the projects did not have a fully defined project scope for using the MUST framework. The following year, a version of the business plan was defined that was based on summaries of the strategic inline analysis definition of the MUST project management to create an overview for coordination of on-going and upcoming projects. In addition, in their final version the business plans were complemented by action plans written in an online document format, as illustrated in Figure 9.8. All the story cards developed would be added to the action plans by the IT-professionals, and the development progress would then be a re-occurring agenda item of the IT-steering committee. The action plan would include the heading of story cards, the time frame of their development, their status, and the IT-developer in charge. The action plan was both used for reporting purposes and for IT-steering committee members to be able to provide input in regard to, for example, coordination with other projects.



Action Plan ICT & Security

4. WMU Web and Portal Solutions

Revise Content Sections - Development, Academics, Admissions - Inspect and correct Gateway settings	Q3 2012	Pending	Erik
Development - Updating Base Platform - Revamp Research Section Layout - Implement Gateways - Create specific RSS Feeds - MoU listing - Configure & deploy Solr Faceted Search - Improved News listing - Revised publishing approval workflow	Q3 2012	Ongoing (individual releases upon completion)	Erik

Figure 9.8 Action plans as part of the business plans

9.2.3 New IT-steering committee processes

The above account of the implementation of the business plan, the MUST framework, and story cards provides a picture of how local project development continuously interacted with the IT-steering committee. The improved IT-steering committee agenda and decision processes were initially prompted by the turmoil of the failing IIS plan accounted for in section 9.1.2. Further changes were implemented following the first business plan, usage of the story cards, and the appropriation of MUST procedures through the three IT development projects presented above.

As was described in chapter 5, the IT-steering committee had for a long-time been an established entity at the university that "drove" the organizational IT management. The previous chair of the IT-steering committee Hughes, for example, described how the committee had a primary role in setting objectives and developing a budget:

Hughes: [The network administrator] and I would sit down and create the budget [inaudible] in the computer committee, and we discussed it, see that it fitted in with the objectives that we set for ourselves
Bolmsten: But was that basically for yourself to work out, what kind of budget that you needed
Hughes: Yes, yes
Bolmsten: And what you wanted to spend it on?
Hughes: Yes, and then it was all itemized, when I presented it to the president it was all itemized, we need twenty new computers in this lab,

However, given the new landscape of increasingly complex and interlaced infrastructure development, its procedures needed to be additionally developed. In addition to fitting computer labs, the IT-steering committee had to coordinate the type of integrated course administration and comprehensive ERP system development described in this and the previous chapter. The chair of the IT-steering committee Laine described how the new processes (1) enabled the committee to focus on the "subject matters" and (2) how they contributed to the existing procedures that were based on the university's rules and regulations' framework and consolidation of best practices from other committees:

"If we are going to work with this, we are going to work with the subject matters, …What I essentially was after also in the computer committee, and we have it partly built into the system, it is in the rules and regulations, there is a standard for how protocols should be written, and that is the beginning, because then everybody can understand, it is easy to comprehend information in a way."

9.3 CMD 2 phase 3: A locally accountable organizational IT management

As the changes described above were implemented to address the need of the shop floor development constituencies for coordinating IT development on an organizational level, the evaluation is based on the perspectives of the different stakeholders voiced in workshops and interviews.

9.3.1 Power from and to the people on the shop floor

The changes to the organizational IT management kept the competence of the management of the infrastructure with the shop floor development constituencies and cultivated their capabilities in also addressing the infrastructure's technical and organizational integration. The recognition that the design of organizational IT management needs to take its starting point with the users on the shop floor was an - at times - difficult but necessary lesson to learn.

The cancellation of the efforts to implement a top-down approach to IT management described in section 9.1.2 is the most prominent example of the power of the shop floor IT management. The shop floor development constituencies gradually withdrew their support and as a consequence it failed.

Another example concerning Administrative Assistant White was developing the electronic forms and the database and her membership in the IT-steering committee:

Dittrich: They have something like where they coordinate. Johan [Bolmsten] told me about it. He told me first of all the steering or the coordination group and then he told me also there is a computer committee?

White*: Why do they have a steering group and a computer committee, what is the difference between the two?*

Dittrich: I don't know.

White*: Me neither. I have no idea, I don't know about that.*

[...]

Dittrich: - and the computer committee and that is still to coordinate the ...?

White*: Yeah, infrastructure, yes, and the budget [...] but bugger that*

[...]

For a while Administrative Assistant White was replaced in the IT-steering committee, as she was not recognized as an IT-professional. This was not only a vulnerability for the EUD that she carried out but, by extension, for the organization, as her expletive, "bugger that" indicated her indifference to bureaucratic formality and she continued with her development - anyway - to the best of her capabilities.

Her example illustrates an important dimension in how the situational and historical setting at WMU of shop floor development constituencies interplayed with the development of organizational IT management. The power in managing the development of software support was not *given* to the shop floor users by management or by IT-professionals, but developed over time based on experience and competencies.

9.3.2 The only partial reach of organizational IT management by IT-professionals

The analysis of the empirical material consistently shows that the reach and mandate of the IT-professionals to manage the technical infrastructure development were only partial. Especially, the two cases of EUD reported in section 9.1 indicate that shop floor users might not even depend on IT-professionals to develop their IT-infrastructures. With the availability of more and more EUD-friendly development tools, the viability of shop floor users to develop their own technical infrastructure can be expected to increase. Organizational IT management today and tomorrow cannot assume that IT-professionals will own the technical infrastructure development agenda.

9.3.3 A deliberate work in progress

The organizational IT management's use of the extended MUST method and its linkage with the story cards, the business plans, and the improved procedures in the IT-steering committee came about in an evolutionary manner. This evolution used local projects as probes to gradually understand the different needs of organizational

IT management. Different shop floor development constituencies also had different needs with respect to organizational IT management:

The further development of the academic administration right from the start needed better IT management. The development was growing to include an increasing number of users across different departments. To answer this need, the business plans from the beginning entailed a comparably extensive use of the MUST method and coordination with the IT-steering committee. In total, the new organizational IT management coordinated seven in-depth analysis workshops with eight faculty stakeholders that included both professors, lectures, and faculty-assistants; the in-depth analysis was then used as an input to eight overall story cards that each evolved the EUD functionality of the academic administration support. The story cards were created together with users on the shop floor in agile design and implementation phases and included both technological and work process changes.

At the same time, the current state of the development of the electronic forms required a different type of attention in the early business plans. The shop floor IT management by Administrative Assistant White that was already taking place was now to be supported by an IT-professional. Here, it became important to properly recognize her development efforts and competencies in the business plan, rather than implementing the full MUST framework.

The organizational IT management needs to maintain flexibility in order to accommodate the needs of different projects. This kind of ‘tinkering’ and radical learning, where the organizational IT management evolve together with the experiences gained with software development ventures allows keeping the organizational IT management in synch with the needs of the organization.

9.3.4 Organizational IT-management with users on the shop floor

With the IT-steering committee, a partial organizational IT management based on representative management principles already existed. However, there was a need for process improvements to make it work in a landscape of increasingly complex technical and organizational infrastructure development. In a retrospective interview, the chair of the IT-steering committee described how the improvements resulted in an organized and constructive approach to planning by focusing on the “subject matter”:

"... then one has the subject matter, one has a presentation, the one who has prepared the case then has to focus on what is suggested[...]. It is important that opinions can be put forward, subject matter arguments, and that it is documented, then that goes a long way[...]. If one can come to a clear concrete decision, and if I then don't get a hearing for my view then one kind of has to accept, there has been a forum, I have put forward the arguments, and they were not approved, then one has to accept the vote of the majority."

It is important to recognize that negotiations of weighting, for example, different interests against each other in the business plans were sometimes difficult. From time to time, as further described by the chair of the IT-steering committee, the discussions in the IT-steering committee were heated:

"The first indication of personal attacks amongst the committee members there will be a yellow card; I took [sic] out a yellow card, the first indication of a personal attack on me, there will be red card."

Although there was never any football type of referee cards used, the citation illustrates that the decisions in WMU's IT-steering committee are actually important. It also shows that its procedures need to be carefully designed in order to support the development in the local projects.

9.3.5 Meaningful artefacts

One of the challenges of implementing a participatory approach to organizational IT management is the need to represent complex technical and organizational dependencies in an understandable way for shop floor development constituencies.

Early inspiration came from the reflection papers used to discuss the situational context of infrastructure issues. They were based on a concrete issue experienced by users on the shop floor and intended to present relevant issues that needed to be taken into account when designing for the local projects. This idea was extended during the action research by using the story cards integrated with the MUST framework and the business plans. It then became the focus of the third action research cycle in-itself, as a described in chapter 8.

This type of representation can be contrasted with the ones provided in the cancelled ISS plan (section 9.1.2), where there were no meaningful representations for deliberations available to shop floor development constituencies.

Even though the representations developed for the organizational IT management need to be regarded as work in progress, they have proven to provide a basis to discuss and decide difficult design questions involving both IT-professionals and domain experts.

9.3.6 Trusted representation

It is important that members of the organization informally and formally can trust those representing them. Mutual trust underpinned the research from the beginning, but only became a central topic in a retrospective interview with the Senior Registry Assistant and a Students Social Officer.

Especially in comprehensive developments of new software support, it is important to recognize that normal day-to-day work does not stop just because an IT development project commences. A busy period when handling a new student intake was used to exemplify how it was not possible for all registry staff to attend vendor presentations and workshops concerning a new part of the infrastructure. However, as Student Social Officer Evans noted, she felt well represented with respect to her needs:

"So in a way, I felt comfortable that [the Registrar] went to this presentations, because he would be able to represent us."

In a comprehensive project, staff in a particular domain need to be able to trust their co-users to represent their interests, in this case in the IT-steering committee. As there are other tasks to do as well, the whole development constituency cannot participate all the time.

Based on the empirical material, we can observe that (1) this trust is often built up over time through the joint work in shop floor development constituencies, and (2) users on the shop floor need to be able to influence those who represent them. Although the latter was understood in the action research, representations were one of the dimensions that were not explicitly discussed.

9.4 Limitations of the research: integrated governance and financial planning

In addition to the improvement reported above, there were also limitations that became visible in the evaluation of the empirical material in regard to organizational IT management. These became visible in the context of the third action research cycle about PD tools and techniques for the ERP system as infrastructure reported in chapter 8. They pertain to how organizational IT management integrate with governance on the wider organizational arena financial planning and are described below.

9.4.1 Integrated governance

The more different the development ventures grew in size and required integration, the more the necessity for the organizational IT management to integrate with other management areas in the organization became visible. It is possible to find support for this fact throughout the empirical data collection and it can be exemplified in regard to both design and administration capabilities of the organizational IT management implemented:

- **Design:** A successful project outcome could be dependent on new positions descriptions, new integrated work processes, and even the creation of new organizational positions. All this was outside the scope of the IT-steering committee in-itself to manage and defined procedures and process to the right instances in other parts of the organization were lacking.
- **Administration:** This was also a fact in the IT-steering committee's ability to overview and manage certain complex workflows. One concrete example that illustrates this issue is the payment of salaries. The payment of salaries builds on a complex web of integrated transactions with high requirements on accuracy and accountability. This entails different cutting surfaces and control points between finance, HR, and operational functions that are, in turn, monitored both internally (by multiple bodies to control for example pension schemes) and by external bodies as auditors. There is no room for these transactions not being granularly defined and integrated, as this immediately would result in salaries not being paid or wrongly being paid. This is a dimension that was not completely possible to design nor manage in the IT-steering committee.

9.4.2 Financial planning

Financial planning is another area where there was more work to do in order to argue for the maturity of a sustained PD approach in the organization. The need for improvements here was noticeable already early in the operations of the computer committee, pre-dating the action research. An example is the interview with Hughes where he describes the partially ad hoc and informal way how the computer committee's budget was approved by the president and the management body. Normally, the computer committee and later IT-steering committee got a certain fixed budget allocated – that was index increased annually. If there was any particular investment that had to be made, Hughes described how he usually got what he wanted, but that it was in part a chaotic process where he could also be ignored without really knowing why. To this end, informal channels came into play, for example, via the president's secretary "I would obviously make sure that the things that I really wanted to achieve was well explained, so when [president secretary] went back she could say why we wanted to do things, which would help our cause."

Another strategy for the computer committee to push changes was to underspecify or tinker with the budget process. Professor Hughes, for example, described:

Hughes: Coming up was the fear of how the year 2000 --

Johan: mmm…

Hughes: … and I said to [the president] at the time, that we will need money in case there was a problem, and so I think I got 300,000 dollars out of him for 98 and 99 to change computers and prepare for the worst if the systems crashed; that didn't happen of-course, that was a big [laughter]."

It was not within the primary realm of the empirical data collection to track this dimension. However, when it comes to special investments of computer and software development with strategic implementation, the possibility of this kind of approach is unsatisfactory in both budgeting and follow-up auditing.

9.5 Summary

The action research deliberations resulting in: (1) the computer / IT-steering committee deliberating organizational IT infrastructure development between management, users, and IT professionals; (2) business plans that relate the individual projects to the overall development of the organizational IT infrastructure; (3) a project model based on a PD method (MUST as developed by Bødker, Kensing, & Simonsen, 2004) extended to comprise the implementation as well; (4) and the IT-coordination group coordinating development from a technical perspective show how it is possible to involve users in a meaningful way in organizational IT management.

With this chapter, the outcomes of all empirical research of this thesis have been presented. This chapter has reported on the most comprehensive CMD research cycle in this PhD research, which also links to empirical research presented in the previous chapters. The improvements to organizational IT management connect to extending the shop floor IT management at WMU (chapter 5 and 6) when facing an increasing need for an integrated technical base (chapter 7). Together, the empirical research give evidence of how a shop floor IT management approach can remain a core capability, given increasingly complex technical and organizational infrastructure development. In this way, the action research reported was not an effort to abandon the existing shop floor management – in favor of, for example, EA – but to nurture some of its core qualities, such as a locally anchored development style that gains its momentum from the domain experts themselves. In other words, it aims at creating an organizational IT management on an infrastructure level that is locally accountable to users - in the same way as they can participate in the management of their local shop floor development projects through, for example, the MUST method (Bødker et al., 2004). In the following chapter the combined results of the empirical research are discussed together with related work.

10 Shop floor users sustaining Participatory Design in the organization - Discussion

In answering the research question of how shop floor users in the organization can participate in the evolution of infrastructure, this chapter discusses five contributions with respect to related work. The first contribution is about how shop floor IT management is an important base of infrastructuring to sustain Participatory Design (PD) in the organization. In shop floor IT management, users together with IT-developers – either in the form of IT-professionals or end-user developers – develop software support that is closely linked to their work. This can be related to as a core innovatory development capability in creating useful software support in an organization. The second contribution is that as software support expands beyond local applications to include technical and organizational infrastructure, coordination of the development of the technical base needs to be included as a subject of PD. Technical infrastructure considerations come to matter and the affordances of the technical base become an important denominator in participatory development between users on the shop floor and IT-developers. The third contribution is about how neither a "drift" nor "control" stance of infrastructure management is satisfactory. Instead, users on the shop floor need to participate in the IT management of both the development of their local software and how it is interlinked on an organizational infrastructure arena. The fourth contribution explicates four interlinked improvements put forward to this end: (a) participatory structures, (b) participatory and evolutionary project management, (c) plans for organizational coordination, and (d) representations suitable to make infrastructure development subject to PD. Extending shop floor IT management to include organizational IT management enables sustaining PD on an organizational level. This also includes the appropriation of PD tools and techniques that enable users and IT-developers to talk about the design of technical and organizational infrastructure.

The final section relates to "infrastructuring” dimensions and challenges thereof when users manage the development of sustained PD in the organization. A number of areas for future research areas are addressed as to how shop floor users can take responsibility for their software support, as it expands beyond their local shop floor IT management to include shared technical and organizational infrastructure.

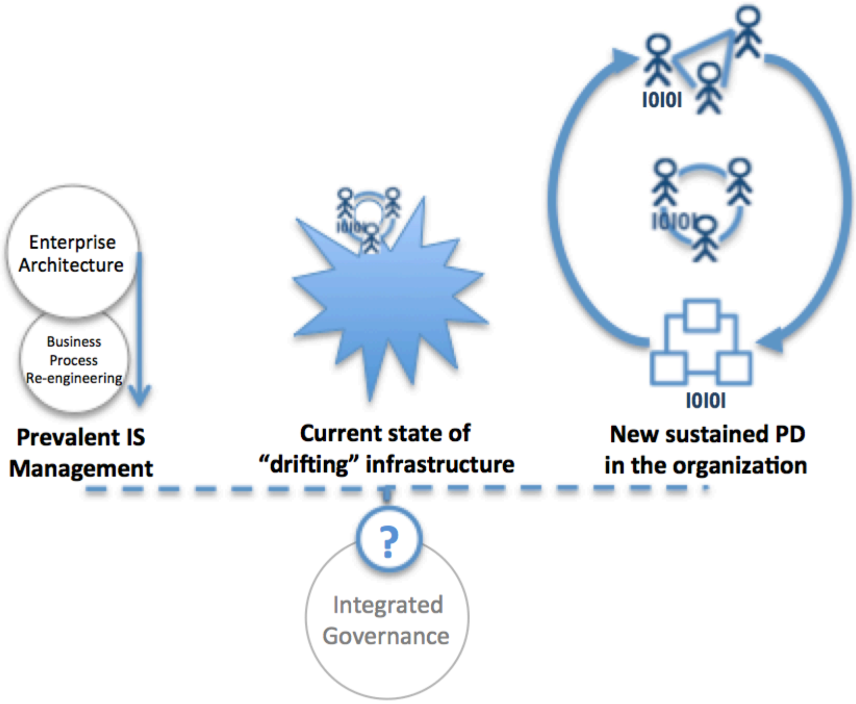


Figure 10.1 Three approaches to organizational and technical infrastructure development and local project development: (1) Prevalent IS management; (2) the current alternative where infrastructure is "drifting"; (3) the empirical results of this study.

Combined, the results of the empirical research contribute to answering a call for a new wave of PD research that reclaims the organizational IT management arena (Björgvinsson, Ehn, & Hillgren, 2010; S. Bødker, 1996; Clement and van den Besselaar, 1993; Dittrich, Eriksén, and Hansson, 2002; Simonsen and Hertzum, 2012) beyond the local project. The action research has supported shop floor users in "infrastructuring" for a participatory organizational IT management to sustain PD in the organization (see right option in Figure 10.1). The infrastructuring perspective in essence turns prevalent Information System (IS) management conceptualizations of infrastructure development upside down. Common IS Management approaches to organizational IT management, such as Enterprise Architecture (EA), are based on a top-down relation to the local project arena (see the left option in Figure 10.1). Instead, the design of working software tools at the shop floor is put in the center, which needs to be supported by a purposeful technical and organizational infrastructure – not the other way around. Infrastructuring in sustaining a PD approach in the organization becomes a process that combines a focus on situated action on the shop floor with participatory planning both in the local project and the organizational level. This combined focus on situated action and planning in this research can be related to from Suchman's (1987; 2007) "plans as resources for situated action."

This, in turn, implies that management is re-positioned on the organizational IT arena to complement the current state of research in regard to "drifting infrastructures" (Ciborra, 2000; Hanseth & Braa, 2001) (see middle option in Figure 10.1).

The empirical findings do not position a comprehensive framework – a valid question is if such is called for or even possible in a PD context. Instead, the discussions of the empirical findings are intended to stimulate discussions among research and practitioners.

10.1 Shop floor IT management as the driver for sustained Participatory Design

A shop floor IT management approach is useful to take advantage of the innovatory capabilities of employees working in an organization to develop software support. A redline throughout the empirical results is how users on the shop floor take responsibility for the development of their own software support, as well as an increasing responsibility for its technical and organizational infrastructure. This extends the original conceptualization of shop floor IT management in Eriksén (1998) regarding users conducting development together with IT-developers – either in the form of end-user developers or IT-professionals. A basic benefit of this arrangement is that software is developed in close connection with the work activities that it supports. Another related and purposeful organizational feature that is shown through the empirical results in this study is how shop floor IT management by nature can be characterized as a hybrid between formal and informal organization. In addition, when taking on increasingly complex development, shop floor IT management enables a mutually beneficial exchange between IT-professionals and end-user developers in their respective professional activities.

In this research, shop floor IT management comes to have an extended role for the forthcoming discussions about sustained PD in the organization. When facing an increasing need to have an integrated technical infrastructure to develop better local applications, shop floor IT management becomes a driver for the development of a sustained PD in the organization.

10.1.1 A core capability for innovation

The organizational importance of shop floor IT management (Eriksen, 1998) is strengthened by the empirical research in this thesis. Primary examples from both the ethnographic and action research show how socio-technical systems that are the result of shop floor IT management are core capabilities in the organization.

This includes the academic administration systems for the registry and the faculty, whose developments have been followed throughout the empirical research. The account of their development begins in chapter 5 regarding shop floor IT management and its rationale in the organization and then extends to chapter 9 relating to organizational IT management. As described in regard to, for example, the registry system (chapter 5 and 6), this is not only a core system in the organization, but has also mandated radical socio-technical innovations on the part of the registry shop floor development constituency. As Registrar Davis recounts, there was no accountable academic management system for the registry when he joined the organization, and due to the special status of the university, a registry system could not be acquired based on a standard model. Instead, it had to be invented in-situ in a continuous joint effort by the Registry staff. Registry Assistant Magnusson and Student Social Officer Evans give complementary accounts of the nature of this development. In chapter 5, Mangnusson, for example describes how shop floor IT management and the End-User Development (EUD) on the part of Davis is an almost implicit part of her work reality and of how working system support is developed: "[…] I have not thought about it before, but now when we are talking about it, it is pretty great […]".

What is further important is how it is shown that shop floor IT management of core systems is not a one off instance. On the contrary, many separate shop floor development constituencies can be found that have used different situated shop floor IT management approaches in developing different types of systems (this is further discussed in the following sections). This makes it meaningful to acknowledge shop floor IT management as a comprehensive and foundational organizational approach.

The function of the shop floor IT management at WMU confirms the shop floor IT management that was accounted for in Eriksen (1998) and Dittrich, Eriksén, and Hansson (2002) of individual developments of system support. There are also resemblances to the cases put forward by Karasti and Syjänen (2004) and Karasti and

Baker (2004) of infrastructuring, in that the shop floor IT management at WMU also takes long-term socio-technical responsibility over its work domain, including both existing systems as well as the development of new ones.

An added multi-relational and processual infrastructuring aspect of the WMU shop floor IT management, which was supported by the action research, was the heterogeneous socio-technical characteristics of their respective developments: The shop floor development constituencies at WMU had different work functions and therefore different types of technical support that needed to be integrated, leading to the development of a common technical platform. For example, the academic management of the registry and faculty had different (although overlapping) functions that needed to be integrated. This technical integration was in part an option that had become available by the advancement of technical platforms. As is further discussed in the forthcoming sections, this required new innovations on the part of the shop floor IT management, both in terms technical and organizational infrastructure. In this way, shop floor IT management shows its continuous relevance as base in the development of core systems in the organization.

10.1.2 Shop floor IT management with IT-development and use competencies

An overall success criterion of the type of shop floor IT management as developed at WMU was that its members had both IT- and use-competencies. The shop floor development constituencies at WMU both included users and IT developers – either in the form of IT-professionals or end-user developers taking on an extended technical development role.

To develop useful software support, the original formula of shop floor IT management at WMU was that particular end-users took on an IT-development role themselves, which extended from appropriation to platform development. In the empirical material, this became manifest in the shop floor development constituencies around the academic management system for the registry and the electronic forms and address database, as described in chapter 6. It was also the case with the development of the first academic management system for the faculty, as described in chapter 7. In the pursuit of keeping the management of IT close to the shop floor, when the shop floor IT management was complemented with IT-professionals, the IT professionals were locally embedded with the end-users on the shop floor.

This mixing of professional roles extends the WMU shop floor IT management from the traditional conception of a Community of Practice (Lave and Wenger, 1991) – which is referenced as a base both in Eriksen (1998) and Dittrich et al (2002) as well as in Karasti and Syjänen (2004) and Karasti and Baker (2004). In line with cases of this PhD research, the users and IT-developers managing IT in a particular shop floor development constituency were a community who on an ongoing and long-term basis deepened their knowledge and expertise of the socio-technical issues in the their application domain. However, the communities did so from the perspective of different professional roles and identities, associated with the domain of use, on the one hand, and IT-development, on the other. In this sense, they can be referred to as Communities of Interests (Fischer, 2001), where the members have heterogeneous expertizes that add value to the community.

The organizational benefits of this purposeful arrangement of IT-development were described in, for example, chapter 7 by Vice-President (Academic) Wang in regard to how IT-professional Nilsson was placed in the same office with the faculty support staff: "*I wanted, this online marking, distribution of marks, but he came up with a product which goes beyond my expectation [...] because he saw our three secretaries, every week."* A feature of this design was also that IT-professionals were asked to take on certain domain specific work responsibilities, and users were encouraged to take on certain EUD tasks; hence, deepening their mutual understanding of each other's expertise and cultivating community building.

IT-professionals and users joining together in design and implementation development activities is per se not a new idea. In design, it is a core feature of PD tools and techniques. In implementation, it is featured in programming-centered development approaches that build on close user-interaction. It is, for example, a feature in extreme programming (Beck & Andres, 2004) - that was used in chapter 9 to complement the MUST method (Bødker, Kensing, & Simonsen, 2004) to create a PD and implementation approach in the local project - through the design practice of having an onsite customer. What is new with the type of shop floor IT management positioned from WMU is to have IT-professionals and users permanently joined. This design also impacts the use of PD in, for example, the need to use tools and techniques to mediate design between users and IT-professionals, and even the need to work with project management in local application development.

This fundamental design of shop floor IT management comes to strengthen a "located accountability" as called for by Suchman (1994, 2002), where use and the development of IT support is closely bundled. It also provides a basis for understanding the forthcoming discussions of infrastructuring in the evolvement of technical and organizational infrastructure.

10.1.3 A hybrid between informal and formal organization

An added strength of a shop floor IT management model that was developed at WMU is that it shows how a hybrid between informal and formal organizations is possible.

As described in the related work chapter, the joining of informal and formal organization is a development related to the notion of Communities of Practice, which can be related to in terms of the "double-knit" knowledge organization (Wenger, McDermott, & Snyder, 2002).

In relation to the empirical cases at WMU, the overall objectives of the developments taken on by a shop floor development constituency were, for example, decided and framed through formal organizational IT management entities; the primary ones being the IT-steering committee (this is further discussed in the following sections) and the departmental management. In addition to being accountable to the respective communities on the shop floor where the software is used, the shop IT management was accountable towards these bodies as well. In, for example, both the cases of Nilsson's academic management development and White's development of the electronic forms and address database, the overall objectives and framing of the developments were organizationally coordinated. The respective shop floor IT management was then up to, in this case, the IT-professional Nilsson and the end-user developer White. Having been tasked to develop the electronic forms and the address database, it was White, herself, that formed and managed her shop floor development constituency with other users when she needed to have input to her IT-development. In Nilsson's case, as part of the organizational framing of his developing activities, he was placed in the same office as the faculty support staff with the intention that they should tightly collaborate. How they collaborated and took on the development of the academic management system was in large up to their shop floor IT management.

This way of management can be discussed in terms of what Wenger et al. (2002) denotes as a "cultivation" approach that diverts from traditional management conceptions in that it is more about electing and fostering participation of communities than planning, directing, and organizing activities. The joining of informal and formal organization, illustrated through the shop floor IT management at WMU, opens up for a new conceptualization of management in organizations. It gives recognition of shop floor IT management that goes beyond informal organization. As is discussed in the forthcoming sections, this is an opportunity when it comes to technical and organizational infrastructure development, but it can also become a vulnerability.

10.1.4 Exchange between end-user developers and IT-professionals.

With the shop floor IT management approach, a mutually beneficial exchange between end-user developers and IT-professionals is facilitated. This is another opportunity of shop floor IT management related to those discussed above where end-user developers meet IT-professionals and where the informal meet the formal.

In regard to IT-professionals benefiting from the expertise of end-user developers, the different episodes of the empirical research shows how end-user developers care about usability, and how they are confronted with the problems of unusable software. In this way, their expertise is also useful to IT-professionals when working with IT infrastructure tasks: As members of the user community and in their capacity as shop floor IT managers, end-user developers were, for example, able to help recruiting the right people for user participation and to prioritize between issues - for example, issues leading to users refusing an application vs. ‘good to have’ features that could wait until IT-professionals had time. A concrete example of this is the comprehensive guidance that Registrar Davis as a long-term end-user developer of the registry offered this researcher as an IT-professional when entering the ERP system development (chapter 8).

At the same time, it becomes visible how end-user developers are in need of the expertise of IT-professionals. End-user developers need to dispose of participatory tools and techniques themselves to cooperate with other users when taking on technical development that goes beyond personal adaptation. Both Davis and White, as end-user developers for the registry system, respectively, electronic forms and address database, had established practices for tasks that are normally associated with professional IT development, such as involving different constellations of users in development. It can be recognized that guidance in terms of participatory tools and techniques to support this type of cooperation is not readily available today. In the action research for this thesis, this was an emergent development in the use of participatory tools and techniques positioned through the MUST knowledge framework (Bødker et al, 2004). In both the further development of the academic system and in the ERP system reported in chapter 8 and 9, end-user developers joined the IT-professionals in working with the participatory tools and techniques. However, this is also an area that is in need of more research as common participatory tools and techniques, as those positioned through the MUST knowledge framework (Bødker et al, 2004) primarily presume that the knowledge divide to be overcome in design is between users and IT-professionals, and that IT-professionals manage the process. In regard to end-user developers also taking on infrastructure development knowledge areas, the cooperation needed looks different due to the changed development dynamics and different competency profiles that EUD and end-user developers imply, compared to professional IT development.

10.1.5 Users triggering infrastructure development

Shop floor IT management becomes both about an ability to continuously develop on an infrastructure as well as triggering and participating in changing the infrastructure in-itself. The type of change to infrastructure that shop floor IT management engage in can in this way be related to as bricolage as radical change (see Andreu & Ciborra, 1998, in related work).

Changes possible through bricolage are similar to the virtues of the type of community PD described in the two cases of infrastructuring featured in Karasti and Syrjänen (2004) and Karasti and Baker (2004). As is also described above, this type of on-going and situated growth of the socio-technical infrastructure that is based on a strong community identity - taking responsibly for the development of the community's interests over the long-term - were key for the shop floor development constituencies at WMU as well. This is the raison d´être of all the shop floor development constituencies and the base of the shop floor IT management in the empirical material of this PhD research.

The other type of change entails more radical changes. What the empirical material shows is that radical change does not stand in opposition to bricolage and community-based development of the infrastructure. In fact, radical changes have been enabled and grounded in the situated realties and in the ongoing developments on the shop floor. The empirical results of this thesis shows how shop floor users are not only content customers of a given technical base provided by IT-professionals, as a meta-design framework (Fischer, 2001). Instead, they also engage in the design deliberations of the technical base. This 'shop floor'-influence to change the infrastructure in-itself happened both in planned and unplanned ways. A primary example of radical changes to the technical infrastructure triggered in an unplanned manner is Levy's critique of the first academic scheduler described in chapter 7, which was canalized in the first reflection papers and shared in the organization to prompt change. Although this example of unplanned infrastructure change, where a single user can impact an entire infrastructure is valid and should not be disregarded, radical infrastructure changes that were situated on the shop floor happened in planned ways as well. As is accounted for in the following sections, with the development of better PD structures, processes, and representations on the organization IT management arena, it was possible for users to take on increasingly more comprehensive changes. Such changes to the infrastructure ranged from the academic management platforms (chapter 7 and 9) to the ERP design project (chapter 8).

That users can participate in infrastructure development through bricolage as well as in radical changes is important for a sustained PD approach; as for an organization to be able to sustain over the long-term, both types of change can be recognized as being necessary.

10.2 The technical base affects participatory infrastructure development

To coordinate the development of a technical base prompt the application of PD beyond the local project. Shop floor development constituencies at WMU triggered the integration of their local software support through a shared technical base. Although the shop floor development constituencies at WMU had different work functions and different types of technical support, there were local synergies to be gained through operating on the same technical base.

This technical integration is in part an option that is made possible by the advancement of technical platforms. In this way, WMU is facing the same opportunities and challenges as in many of today's organizations where single applications become more and more part of joint technical infrastructure supporting cross-organizational and sometimes inter-organizational cooperation. The necessary standardization can be expected to impact the freedom for specific adaptation and development on a local level (Hanseth, Monteiro, & Hatling, 1996). In addition, it is discussed below how the technical base was found to frame the applicability of PD tools and techniques in working with users.

From an infrastructuring point of view, the technical platform has to be selected carefully to provide the possibility to interface to heterogeneous applications, and to allow for non IT-professionals to use it for a base for EUDs. In addition, in the context of the above discussion, evolving and introducing new technical platforms also impact end-user developers taking on technical infrastructure development themselves. Both shop floor users and end-user developers, in this way, need to be involved in the design process and provide the necessary support to update their technical proficiency.

If PD was going to be organizationally viable according to core principles of user influence, the technical base could thus not be black-boxed in the design deliberations with shop floor users. This contrasts with the major IS management approaches, where the technical base is predominantly a management concern, not a shop floor concern. The empirical results of this thesis thereby confirm Hanseth and Braa's (2001) critique of the reward of a shiny casket of gold for the one who manages to come up with universal and pre-set infrastructure standards remains, although maybe persuasive, still an illusion. It also, however, contrasts many second generation PD research contributions where the focus has been on tools and techniques in the local project, and where the technical base has been argued to be outside the primary concerns that PD developers need to deliberate with users (Kensing, 2000).

The contribution of the empirical results that are discussed below are not about accounting for the standalone technical programing languages or technical architectures of, for example, HTML, ASP, JavaScript, ASP.NET, SOOP, and DotNetNuke that were featured in the empirical material. What the empirical results instead focus on is how these technologies come to matter in a socio-technical relation in "infrastructuring." What is of relevance in the design with shop floor users is what parts of the technical infrastructure they use and how they use it - what Pipek and Wolf (2009) relate to as a focus on "work infrastructure." At the same time, what is also highlighted in the discussion of the empirical results below is how the characteristics of different technologies are not just an implicit variable in design, but directly affect how PD design methods can be applied between people on the shop floor and IT-professionals and the resulting design outcome of software support.

10.2.1 Technical infrastructure underpinning local shop floor development and vice versa

The empirical results of this study show how the technical base plays an active part in infrastructuring. The dynamics between the common infrastructure enabling and, conversely, reacting on local development as developed by Hanseth and Braa (2001) is visible in the empirical materials here as well. The empirical material additionally shows how the standard set by the technical base and the infrastructure developed on top of it confines local technical development. At the same time, emerging technical options that are situated in local development push the development of the infrastructure standard.

In the analysis of the technical base evolutions in chapter 7, the different traits of the technical base illustrated how the design of certain functionality was both enabled and constrained (as well as design practices, as discussed in the following section): the first ASP evolution provided flexibility for custom development (at the expense of being time-consuming to program); whereas the second ASP.NET evolution opened up for improved API access to integrate with other applications used; finally, the third evolution, where a fourth generation technical base was implemented, came with an extensive catalogue of pre-developed modules, which would not have been feasible to develop in-house but which could be further developed. Also in the empirical materials of the other chapters, the traits of the technical base played a role for the applications that could be developed: for example, the improved integration of the academic management functions that were called for by the registry and faculty department were enabled by the technical base of the ERP system in chapter 7.

Different from the Hanseth and Braa's (2001) case, the dialectics between the technical base standardization and localization did not play out as contradictions between the central IT management and the local shop floor requirements; rather, the shop floor needs were the main drivers to move from a custom development environment to a fourth generation technical base.

10.2.2 The technical base frames the Participatory Design practices

Based on the empirical results, it is claimed that technology not only affords the provisions of technical functionality but also influences Participatory Design (PD) processes. In the traditional perspective on software development, the development process influences the design of the product, which then frames the possible ways of using the software. Similarly, many participatory researchers black-box even the implementation process by emphasizing PD methods and practices for the pre-study phase – see Kensing and Munk-Madsen (1993) in the related work section and its practical explication in Bødker's et al. (2004) MUST method. This is also recognized by Orlikowski (2010) who discusses a tendency within process-oriented IS development approaches that focus on socio-technical engagement, such as PD, to downplay the characteristics of the technical dimension in tools and techniques used to cooperate with users.

The first technical base for faculty academic management that was part of the action research, as reported in chapter 7, enabled the development of highly customized functionality and was geared towards a close design and development collaboration between IT-professional Nilsson, Vice-President (Academic) Wang, and a close realm of faculty support staff. The following technical bases opened up PD to include a gradually increasing number of users and other stakeholders. Ready-made modules that were easier to install and configure subsequently made it easier to prototype and showcase functionality. In addition, due to a consequently decreasing need for programming expertise, the new technical bases also opened new opportunities for EUD. This enabled new socio-technical interfaces not only between IT-professionals and users, but also between IT-professionals and end-user developers. However, at the same time, as, for example, the case of the document management module showed, more standardized technical bases with ready-made modules also made it more difficult to customize functionality. This resulted in frustration on the part of users, where they could showcase in design how they wanted a particular feature that then could not be implemented.

This illustrates that the technical base in place not only affects the design space, but it also influences how the working relationships of technology production and use can evolve. This is in line with the Software Engineering side of the PD discourse that argues that iterative and evolutionary development methods provide a better base for cooperative design, as they allow the users to evaluate partial designs and revise it based on the evaluation. The suspicions among software engineers that technology influences not only the final design but also the design and development processes thus turns out to be justified. This is, however, to date only an emerging discussion both in Software Engineering and in PD. It can be related to the evolutionary PD stance put forward by Floyd (1991) additionally, Simonsen and Hertzum (2013) have more recently placed related recognition of a need for iterative design and development as part of a sustained PD in the organization.

10.2.3 New technical bases equal better Participatory Design?

It follows from the discussions above that the development of new technical bases per se is not a sole denominator of better (or worse) possibilities for PD.

In chapter 7, it was asked if Web 2.0 was the answer to our participatory prayers? Based on the empirical material, it is concurred with Floyd, Jones, Rathi, and Twidale (2007) that the developing capabilities of Web 2.0 carries important opportunities for PD. However, the argumentation of the previous section that different technologies have different affordances can accordingly also be explicated on a technical level in regard to Web 2.0. The empirical findings show that Web 2.0 technologies are both enabling and constraining. High-fidelity prototypes and mash-ups have added new capabilities of how users can participate in the development of software support. Nevertheless, they also have added constraints to custom development. This realization was prompted (chapter 7) in the different evolutions of the technical base for the academic administration. The same was true in regard to the continuous academic management development design of the new ERP system reported in chapter 8. The empirical results of this PhD thesis thus provide evidence that new technical bases do impact PD, but that they are not synonymous with one-way destined, improved opportunities for PD; consideration has to be made of what type of PD is sought after.

10.2.4 The technical infrastructure links the IT support of separate work practice ecologies

The technical base as infrastructure links software supports of different local work practices. The document management module and its usage in the faculty academic management part of the infrastructure reported in chapter 7 provided this insight that is not found to be accounted for in any related work. The document management module was implemented for a number of functions before being used for the academic management. In these cases, it successfully supported the work practices. When working with the academic management, however, the former turned into a strait jacket. This new dimension, where the relation between local pieces of software support need to be taken into consideration when they operate on the same infrastructure, offers yet another reason as to why the technical base cannot be black-boxed in the local design project (Bødker et al., 2004; Kensing & Munk-Madsen, 1993). Another example that follows relates to the design of the ERP system reported in chapter 7, where two different shop floor development constituencies, with different needs and perspectives on academic management work, needed to both technically and organizationally integrate.

Optimally, a satisfactory "holistic" technical solution seeing to different needs should be put in place in these cases, which have to be negotiated outside the individual local project application. However, this mandates that users understand and engage in deliberations regarding the technical base outside the parts of the infrastructure they actually use - their particular "work infrastructure" (Pipek & Wulf, 2009). The appropriation of PD tools and techniques to support infrastructure development together with the engagement around the IT-coordination group and IT-steering committee, discussed below, are examples of possible structures, processes, and representations to negotiate the tradeoffs as well as synergies when different work practices are linked.

10.3 Beyond control and drift

The need of different shop floor IT management to coordinate their development on a shared technical base calls into consideration how to organizationally approach the management of technical infrastructure. Going into the action research, two options were on the table from related work: control or drift. Both were challenged by the empirical results of this thesis.

10.3.1 Beyond control

As the contributions from the empirical results of this PhD research show, a participatory organizational IT management cannot be related in terms of a configuration or instance of predominant IS management approaches. These are not compatible with how the members of the shop floor development constituncies need to manage IT on the organizational arena, and where due consideration is given to knowledge and power diversity. This was thus not an assumption from related work going into the research, but an empirical finding. The failed outcome of the Information Services Stratigic plan described in chapter 9, where the IT manager was employed to organizationally implement a new IS management scheme, also gave evidence to this. This implementation effort was not part of the action research of this thesis, but created a mandate for the deliberations of a participatory organizational IT management from the shop floor. The concerns with the Information Services Strategic plan was that it jeopadized the working function of different shop floor IT management constituencies and that it could not cater for the necessary technical infrastructure that was needed to support the local development. The most vocal critique was voiced by Vice-President (Academic) Wang and Registrar Davis, who had a dual function as users and end-user developers, as well as being managers.

10.3.2 Beyond drift

However, the shop floor people development constituencies were not content with a "drift" approach (Ciborra, 2000) either. They, themselves, called for improvements to organizational IT management. To sustain the established shop floor development practices and the already existing management structures, such as the computer committee, in the light of increasing needs for integrated technical and organizational infrastructure was one of the requests of the organization when going into the action research.

This presented a challenge for the action research, where relevant guidance in related work was limited. This can be seen in the light of Ciborra (2000), where the empirical results in this research, on the one hand, are in line with Ciborra's critique that the necessarily situated characteristics of infrastructure development makes it "drifting" from a top-down IS management perspective. On the other hand, Ciborra (2000) questions the possibility for "meta-decision making forums" (p 39; also see Peppard, 1999) - which is essentially what the IT-steering committee together with the other outcome of the participatory organizational IT management deliberations is about.

An opportunity given by the empirical domain of this research is that organizational infrastructure development could be approached from a different point of view compared to, for example, Ciboorra (2000), Hanseth and Braa (2001), and many of the cases in the 2009 special issue of JAIS described in the related work chapter. In the cases they report, the development of local IT needs are in opposition to already established control-oriented IS management practices in large corporations. In the case of WMU, the conditions and thereby the outcome were different, providing an opportunity to design a new organizational IT management. There was no comprehensive infrastructure in place based on, for example, standardized IS management approaches (Bernard, 2005). And the reason that "drift" (Ciborra, 2000) was not sufficient as an endpoint to understand and deliberate a supportive infrastructure was that shop floor users, themselves, wanted to manage IT on the organizational arena. This type of meta-decision making for technical and organizational infrastructure development may still, from the top-down, be perceived as "drifting," but from the bottom-up can be characterized as useful and purposeful.

The proposed solution to answer the research question of how end-users can participate in the evolution of an organization's IT-infrastructures from the empirical results of this thesis is to conceptually combine "drift" and "control" in accordance to the relation put forward by Suchman (1987, 2002) when proposing "plans as resources for situated action." The situated development of infrastructure is in focus, where organizational IT management – still governed by users – becomes positioned as a resource when the different shop floor IT management constituencies find it useful.

10.4 Supporting sustained Participatory Design in the organization

To answer the research question of how end-users can participate in the evolution of an organization's IT-infrastructures, the results of the action research put forward combined improvements to participatory structures, processes, and representations (tools and techniques) to support an organizational IT management. These improvements are important in order to sustain PD in the organization. The focus has been on how the situated shop floor IT management could be expanded with a participatory IT management on the organizational arena to design and coordinate increasingly integrated technical base development projects. This section discusses the participatory improvements made. The improvements provide further insights to the development of a shop floor IT management approach on the organizational arena. This adds to the original research on shop floor IT management by Eriksen (1998) and Dittrich et al. (2002) with measures needed to maintain IT management with users on the shop floor when local applications are integrated and extended to technical and organizational infrastructure development.

The discussion relates the empirical results to the challenges posed in related work of managing the development of a sustained PD in the organization by Simonsen and Hertzum (2013). As described in the related work chapter, sustained PD is an emerging research focus positioned by Simonsen and Hertzum (2013) - which can also be anchored to earlier PD research such as Floyd's (1989, 1991) participatory and evolutionary system development. The objective is to extend PD to an iterative approach that includes stepwise implementation as well as catering for improvisational change management, including anticipated, emergent, and opportunity-based change. A number of challenges are recognized for future research: creating appropriate conditions for PD; managing a multitude of stakeholders; and managing a stepwise implementation process. These challenges are addressed in the discussion below: the first sub-section puts forward two types of conditions that are of importance when working with measures to sustain PD in the organization; the second sub-section discusses the importance of purposeful organizational structures that give users influence on the organizational arena; the third sub-section is about an participatory and evolutionary project management; the forth sub-section is about planning the coordination between project management and organizational IT management; finally, the fifth sub-section is the importance of participatory tools and techniques.

10.4.1 Conditions for sustaining Participatory Design

Creating appropriate organizational conditions for PD is both an important and potent challenge to address if working to sustain PD in the organization. In this case, the shop floor IT management and how the technical base matters in PD, as discussed earlier, were contingent conditions to support the development of sustained PD in the organization.

Having a long-term strategic recognition and mandate of a shop floor IT management model, as was the case at WMU, was an important condition to the continuous development of PD on the organizational arena. Although the particular characteristics of the situated shop floor IT management as described here are by nature unique to WMU, the occurrence of shop floor IT management as such and other related approaches are not unique. Such related approaches are featured in the cases of Eriksen's (1998) shop floor IT management, Dittrich et al, (2002) "PD in the wild," and Karasti and Syjänen's (2004) community PD. In whatever shape and form they come, identifying and anchoring to them, can be seen as an asset in development efforts by an organization to sustain PD.

Another asset that both motivated and aligned users and management to participate and endorse the development of the participatory structures, processes, and representations deliberated through the action research was to base the development on a real issue situated on the shop floor. In this case, the complexities of an improved integrated technical base provided both a real need and a trigger for organizational coordination of the IT infrastructure from the shop floor. This was doubly beneficial, as both management and users recognized the issue but gave the users a problem formulating initiative. Finding a way to give users influence not only about the design of a solution to problem, but about the problem in itself can be recognized as a benefit in a participatory approach.

In this way, in their own rights, both the existing shop floor IT management approach and the technical base in need of organizational coordination were enabling conditions that interplayed with targeting the other challenges of sustaining PD in the action research, as discussed below. These challenges can be related to managing a multitude of stakeholders and a stepwise implementation process (Simonsen & Hertzum, 2013).

In working to sustain PD in the organization, anchoring to the shop floor IT management of shop floor development constituencies and a need to coordinate technical infrastructure development can provide a path forward for other organizations as well. On a general level, there might, however, also be other situated participatory practices and needs that can be used as an anchoring.

10.4.2 Structuring end-user influence on the organizational arena

Working with organizational structures where end-users are influential is important in managing an increasing number of stakeholders and maintaining a participatory mode of operation in the organization, also beyond the local shop floor development constituencies.

As an enabling condition, the shop floor development constituencies at WMU were capable of planning their own local development of software support. Their situated nature also enabled them to pick up on emerging possibilities and turn them into development opportunities of software support to socio-technically improve their work practices. A challenge was to maintain a similar modus operandi in caring both for a situated and planning dimension on the organizational arena.

The empirical research shows the possibility to connect shop floor IT management to a committee-based IT-management structure on the organizational arena that is grounded in end-users' influence. The computer committee (later IT-steering committee) at WMU was initiated by end-users and was one of the most long-standing management structures in place at the university. It comprehensively coordinated the management of IT in the organization, including managing the organizational IT budget. Such committee-based management structurally provides an opportunity for users to influence strategic decision-making, which is outside their realm of activities in common IS management approaches (see description of IS in chapter 3).

The improvements deliberated through the action research further show that such committee-based management can be developed to take on increasingly complex socio-technical infrastructure development. This infrastructure development entailed that an increasing number of stakeholders needed to participate in decision-making processes, including more and more end-users from different shop floor development constituencies, IT-professionals, and managers. As described in chapter 9, making decisions on infrastructure matters were not necessarily always easy, where sometimes difficult weighting between strong interests was necessary. Hence, although the shop floor development constituencies themselves called for points of integration between their local IT applications, this did not mean that this integration always was straightforward. In the context of the projects described in the empirical chapters, such issues could both entail planning and canalizing on emergent ideas for changes that individual shop floor development constituencies wanted to realize on the organizational arena. To mediate such issues when revamping the computer committee as the IT-steering committee, new robust processes were put in place - that also linked to the project management and representations discussed below. These included preparation of agendas, presentations, and decision processes – where, if necessary, voting was one instrument – and writing of protocols, another. As the chair of the IT-steering committee Laine put it, this was necessary to be able to focus on "the subject matters" in an effective manner. Such subject matters included making decisions on projects that spanned individual shop floor development constituencies, deciding on their organization and planning, and evaluating their progress.

Another structural change that was part of the action research was to organizationally position the IT-coordination group as a resource to the committee-based management. To institute an IT-coordination group to organize IT-professionals as such is hardly revolutionary, and would be a commonsense feature of any standard IT-management scheme. However, positioning an IT-coordination group as a resource at the disposal to a committee-based management where the influence of shop floor users is a primary concern is novel and contributes to understanding how to sustain PD in the organization.

In working to sustain PD in the organization, it was important to work with user-influence in the decision-structures in the organizational arena. The long-term established computer committee at WMU that continuously underwent changes to maintain its organizational relevance - finally ending up in the form of the IT-steering committee - is a solution to this end. In general, its committee-based management structure and mode of operation give evidence that a bottom-up management over the long-term can function in the organization.

10.4.3 Participatory and evolutionary project management

An improved participatory and evolutionary project management is equally important to answer to the challenge of managing multiple stakeholders. A purposeful project management is also important to manage a stepwise implementation in a participatory way. To target both these challenges was especially important when the local shop floor IT management was organizationally expanding.

The participatory and evolutionary project management put forward as a result of the action research was based on a development of Bodker's et al.'s (2004) MUST method to include both PD and implementation. As described in chapter 9, the MUST method was selected as a base because it presents a coherent user-centered project management framework that builds on PD principles such as: a coherent vision for change between IT systems and work practices; genuine user participation in IT-design; IT-designers having firsthand experiences with work practices; and anchoring visions for change with affected user constituencies.

It, however, needed to be extended to include iterative software development and to connect to the overall infrastructure decisions decided by the IT steering committee. The main objective with the development of a new project management approach was to introduce a planning mechanism to the organizational arena that complemented the shop floor IT management, especially intended for projects that spanned across different shop floor development constituencies. In addition, it was of importance to be able to pick up on, and channel, emerging changes from the shop floor to understand development opportunities on the organizational arena. The idea was thus to complement, not replace the shop floor IT management. The organizational mandate for change, therefore, included not adding unnecessary "bureaucratic" red tape, as expressed in a workshop by an IT-steering committee member. In selecting MUST as a base for project management, it was emphasized as being of importance by the Chair of the IT-coordination group how it could be streamlined and continuously adapted to fit projects with different needs and different sizes.

The MUST project management method, as originally defined by Bodker et al. (2004), contains four phases: an inline analysis to connect to strategic considerations, and in-depth analysis to understand situated work-practice requirements in detail, and a design proposal.

Two extensions are put forward by the action research in this thesis: the first extension is to connect the strategic inline analysis of MUST to the development of business plans (also see next section) in order to open up the projects to participatory management on the organizational arena; in this case, it is the committee-based management in the IT-steering committee. The second extension is to revise the design phase to also include implementation. Instead of one final design proposal, an evolutionary design and implementation phase is put in place. As described in chapter 9, this evolutionary design and implementation phase used Beck and Andre's (2004) notion of the planning game with story cards and Kyng's (1995) paper on making representations work as a model. The intention is to "remind" participants of current work situations in need of change and present a vision for overall change, entailing both new system designs and work practice designs. This can also be related to Naur's (1985) notion of "theory building" in software engineering, where the theory-in-mind about the application and its application domain has primacy over project documentation in development. To this end, the primary use of story cards is to support ongoing close collaboration between users and IT-professionals, not to provide a standalone catalogue of project documentation.

The experimentation with this way of planning development began already with the practice of the reflection papers, as described in chapter 7. The idea was not to present a comprehensive documentation scheme to define development, but rather to use representations as boundary objects between different stakeholders. The success of this approach also built on the participatory structures as discussed in the previous section, where developments on the shop floor were closely interlinked with structures in the organizational arena.

The story cards were divided into different parts. The first part of a story card gave an overview of a concrete area in need of change, grounded in experiences from the users' point of view. The second part of the story card then presented a vision for overall change, both regarding technical functionality as well as user interfaces and work organization. The final part of the story card described the technical and organizational implementation. In this way, the story cards represented both the system being designed and the work situations where the system would be used. The number of story cards written depended on the size of the project. Suggestions of new functions through the story cards could continuously be put forward as the project progressed, and were then prioritized by the project members (with possible feedback from the IT-steering committee). As the scope of the projects increased, as, for example, were in the case with the ERP system, described in chapter 8, the story cards could also be combined with other PD tools and techniques, such as participatory observations, to further enhance their function as boundary objects (also see the discussion of representations in section 10.4.5).

The participatory and evolutionary project management as a result of the action research shows that a participatory approach can be sustained also beyond local shop floor IT management when the size and scope of projects increase. The added evolutionary design and implementation component to the MUST method (Bodker, 2004) gives one possible answer to the challenge (Simonsen & Hertzum, 2013) of how a stepwise implementation can be added to PD.

10.4.4 Organizational plans for coordination

A contribution related to the improved project management is a planning mechanism to link project developments to organizational structures and decision-making processes in the organizational arena. Through the business plans, it was possible to maintain a wider organizational accountability in a participatory way when an increasing number of stakeholders also outside the immediate realm of the projects were affected by project development. The business plans were developed as an add-on to the participatory and evolutionary project management. They intended to support a planning function on the organizational arena in working as an intermediate between the projects and organizational structures for planning. The business plans linked to the strategic inline analysis of the project management and were used to create accountability and transparency for the IT-steering committee. They were also updated continuously as the project planning and development evolved. This update consisted of short bullet-point summaries from the story cards. In this way, the business plans show an approach that answer to the challenges of how to maintain communication between an increasing number of stakeholders within and around projects, as well as how to put in place a strategic reporting function for the stepwise development progress of projects.

10.4.5 Participatory tools and techniques for infrastructuring

If users on the shop floor are going to be influential in managing planning and capturing emergent opportunities for change of technical and organizational infrastructure, it follows that there need to be ways to support communication between stakeholders. The empirical results of the action research present a number of participatory tools and techniques that connect both to the participatory and evolutionary project management and structures, as discussed above.

Similar to the participatory and evolutionary project management, the development of participatory tools and techniques started with the homegrown practice of using reflection papers in the first Cooperative Method Development (CMD) action research cycle about a shared technical base. It also became of importance to have representational means that enabled shop floor users, IT-professionals, and managers to talk about technical and organizational matters of infrastructure development in a meaningful way. In line with the representational qualities argued for by Kyng (1995), the reflection papers intended to prompt reflection through reminding - not comprehensively defining - actors of relevant dimensions of infrastructure design. In addition, the design of the reflection papers as representational means of infrastructure matters took a stance in the users' issues. The first reflection paper, for example, used Faculty Assistant Levy's practical experiences of issues with the technical infrastructure as an input to prompt deliberations of improvements.

The reflection papers, in this way, worked in accordance with the guidelines of the Cooperative Method Development CMD approach (Dittrich, Rönkkö, Eriksson, Hansson, & Lindeberg, 2008) in regard to setting a focus on (1) shop floor software development practices taking the practitioner's perspective and (2) deliberating change together with involved practitioners. Initially, in the action research, the reflection papers as representational means for infrastructure development were a situated action and research response to support reflection, and did not constitute a finished concept. To this end, the importance of asking questions related to who has the "modeling monopoly" was described in the empirical material as what the shop floor people actually understand. Did they understand the infrastructure deliberations or did they understand that the IT-professionals understood them?

As a consequence, in the second and third CMD action research cycle, reported in chapter 8 and chapter 9, the systematic appropriation of participatory tools and techniques was targeted. Again, MUST was used as a base with its knowledge framework to position PD tools and techniques in relation to project management (Bødker et al., 2004). As discussed below, the empirical results show how a number of PD tools and techniques can be used to negotiate technical and organizational infrastructure dimensions of an integrated ERP system that spans the local development in several shop floor development constituencies.

Functional analysis: The application of the functional analysis put into the hands of shop floor users as a PD tool shows how an otherwise commonly management-oriented task of integrated project scoping can be pursued in a user-oriented way. As documented by Bødker et al. (2004) in the MUST approach, the functional analysis is a management-oriented technique, where one aim is to identify the work functions of which a design project should focus on (p. 208-209). The value of applying functional analysis as a PD tool has also been documented by Simonsen (2007) in linking the local project to the business strategy of an organization. The ERP design project of this research shows an additional PD application of the functional analysis. When working with the type of complex infrastructure development that an ERP system constitutes, it was important for users and other stakeholders to come to terms with how different shop floor development constituencies that were affected by the project needed not only different amounts, but also different types, of attention. As described in the evaluation of the ERP design project, functional analysis was used for this purpose to support high-level requirements soliciting to negotiate the scoping of cross-departmental developments. This became of importance to allocate resources in the project planning managed by the IT-steering committee.

Rich-picture workshops and participatory observations: The use of rich-pictures and workshops in the ERP design project can be discussed in terms of enabling stakeholders to jointly participate in what Star and Ruhleder (1994) refer to as an "infrastructural inversion": to elevate dimensions of the infrastructure that are otherwise sunk-into socio-technical work realities, and therefore are not otherwise ready-at-hand for reflection. In this way, it was described how the previous social-technical infrastructure design of the faculty and registry academic management system were in need of improvements, but that its design had an historical logic in choices and politics that defined the original shop floor development constituencies. Vice-President Wang and Registrar Davis, for example, came from different academic traditions, where student support and student integrity were related to differently. These core assumptions came to influence their perspective of what an academic administration system was about and made it difficult for them to reflect on new and integrated design options – even though they both were of the opinion that options were needed. Their different perspectives imposed blindness and bounded rationality to envision new design options. This blindness became part of the implicit socio-technical relation of the original infrastructure. The use of rich picture workshops in an infrastructure setting, in this way, supported raising different dimensions of the infrastructure to awareness again for joint deliberations of improvements. They show how PD tools and techniques can work in close concert to elevate complex infrastructure relations for joint reflection. The focus to support strategic design decisions in moving from one infrastructure to another became on communication between stakeholders instead of comprehensive documentation frameworks (as, for example, in Bernard, 2005).

Participatory observations, story cards, and vendor presentations: The use of participatory observations came to illustrate another dimension in terms of giving the users influence over infrastructure decisions. The application of participatory observations in the ERP project shows how it is possible to mediate a design “model monopoly” (Bråten, 1973) on the part of IT-professionals to the benefit of shop floor users. In the choice of ERP systems, it is a known concern even from the vendor’s point of view that it is difficult for shop floor users to have a meaningful input into design choices due to the complexity of the system. In this way, all the “groundwork” - as Student Social officer Evans termed it – in IT-professionals conducting participatory observations and even partly overtaking work functions aided their ability to help the users to formulate their requirements. In this case, it enabled the IT-professionals to support the users in an iterative process to write the story cards that were used for project planning. These were then provided to the vendors to enable them to create presentations that targeted "real" issues. In the ERP project, this concretely enabled the users to have a meaningful input in the contacts with the vendors of the ERP system. Although it took longer before a contract was signed, this arguably enabled a better match between organization and ERP system provider – hopefully recognized to the benefit of both. Participatory observations, as applied here, have a commonly documented function to enable IT-professionals to get a concrete experience with users' present work practices (Bødker et al., 2004). In this way, professional IT-professionals are in need of PD tools and techniques, as well as users. Due to the comprehensive nature and complexities of infrastructure development it was shown how it was valuable for IT-professionals to help users to articulate their design needs. With the use of participatory observations, this articulation support was not based on abstract design ideas of new technological options or "best-practices." In addition, it is shown how it is valuable to think about the application of PD tools and techniques in connected layers: participatory observations enabled IT-professionals to support users in articulating requirements to the vendors, which, in turn, enabled the vendors to make presentations and prototypes that were meaningful to the users.

In general, the use of representations, as discussed above, shows a contrasting approach to IS (Bernard, 2005). Both IS and the use of PD tools contain elements of communication and documentation. However, in line with Kyng (1995), the application of participatory tools and techniques is focused towards communication, whereas the function of representations is to remind stakeholders of relevant dimensions for design, not to define the application domain through documentation frameworks and artifacts. The use of representational means is also centered around the users.

In IS, an ERP system is a good example of an infrastructure that is in need of the type of comprehensive documentation frameworks that are described by Bernard (2005) and exemplified through the EA3 cube. In reference to the above ERP case at WMU, the current and future of the academic administration would, for example, be documented in different hierarchical levels along different lines of business for the faculty and registry. This enables the transition from the current to the future architecture in a standardized and effective manner, where different stakeholders can focus their efforts on their particular documentation layer(s). Users on the shop floor are, for example, not typically expected to participate in development beyond the fixed sub-system of their work practices. Signs were found though of recognizing a different type of use of representational means in Enterprise Architecture. Barn and Clark (2011), for example, cite Naur (1985), arguing that the "theory in mind" has primacy over documentation frameworks. Whether this would give users a prominent role is unclear.

The research results of this PhD research showed how the stakeholders in the ERP design project could collaboratively build up their own theory of the application domain, which had primacy over producing standalone products such as documentation and specifications. What the representational means in this way enabled was for the shop floor users, IT-professionals, and other stakeholders involved to come to terms together with how the ERP project related to different socio-technical issues grounded in their work domains, as well as to explore various new integrated solutions of how modifications could lead to improvements.

10.4.6 Summary

The discussion of the improvements to sustain PD in the organization presents a coherent chain of reasoning: It takes its stance in shop floor IT management as a core capability in creating useful software support in the organization. When software support benefits from being integrated in the same technical and organizational infrastructure, PD is then required to enable design between shop floor development constituencies. Shop floor IT management and integrated software support on the same infrastructure are, in this way, enabling conditions and a raison d'être of sustained PD in the organization. To answer the question of how end-users can participate in the evolution of an organization's IT infrastructures, four interlinked improvements were discussed to PD structures, processes, and representations: (1) the importance of organizational decision-making structures where end-users are influential; (2) participatory and evolutionary project management that supports situated shop floor IT management; (3) a planning mechanism that links different development projects; (4) and the appropriation of PD tools and techniques to support infrastructure development.

10.5 Infrastructuring with Participatory Design

This section provides additional perspectives of infrastructuring in relation to the improvements of participatory structures, processes, and representations to sustain PD in the organization. In the thesis, infrastructuring (Karasti and Syrjänen, 2004) provided an analytical framework to understand how situated shop floor IT management practices connect to supporting technical and organizational infrastructure, which supported the empirical research. In turn, the improvements made through the action research further the understanding of infrastructuring with PD in an organization, and also open up for a number of additional areas in need of more research: It is recognized how acknowledging the knowledge and decision-making of shop floor users on all levels of infrastructure development is needed, but remains a challenge for action and research. Along the same lines, it is discussed that although the organizational IT management positioned in this thesis can be related to conceptualizations of infrastructure management in related work, there are few other examples where such management has been implemented. To this end, the empirical results position a path forward of how IT management in organizations can be re-positioned where end-users are put in the center, which is grounded in related work. It is also discussed how this gives IT-professionals a new role. Finally, the process of infrastructuring in this research is discussed in terms of "expanding circles of located accountability."

10.5.1 Acknowledging the knowledge and decision-making of shop floor users on all levels of development

The relevance of shop floor users' knowledge and decision-making need to be acknowledged on all levels of infrastructuring, ranging from local adaptation and development to infrastructure development.

Pipek and Wulf (2009) approach infrastructuring from a work infrastructure perspective, where in-situ design work of the infrastructure from the user's point of view is the starting point of analysis. The focal point of analysis is thereby not comprehensively all possible dimensions of an infrastructure. However, as the empirical research here shows, this should not be interpreted that users needing to participate in infrastructure development is limited to use dimensions only. Related to Pipek and Wulf's (2009) framework that depicts infrastructural layers of technical development activities, as described in the related work chapter 3.5, users' engagement needs to span "points of infrastructure" of technology breakdowns in-situ design work. Users need to participate in preparatory design work and infrastructure background work dimensions as well.

To this end, what stands out in the WMU case is how the established shop floor IT management practices over a long period of time have ranged across a wide set of scenarios of how end-user developers share their development with their peers, ranging from shared usage, to cooperative tailoring, to shared infrastructure – where the last scenario is the least researched (Pipek in Lieberman et al., 2006).

This was true both in regard to EUD going on in the registry and administrative shop floor development constituencies. However, although both Registrar Davis and Administrative Assistant White had a strong, already established status and recognition in the immediate realm of their technical development in regard to shared usage and cooperative tailoring, they were comparably vulnerable to the last scenario of shared infrastructure development. On the one hand, they developed part of the IT infrastructure for the whole organization and provided important tools. They were aware of their role, consciously including relevant stakeholders, as needed in their development efforts. On the other hand, as the episode with contact database shows, there were issues in terms of how they were officially recognized as IT developers, implicating that other organizational actors might not have been aware of the full range of their IT-development activities. Therefore, if wanting end-users also to take on infrastructure development, it is important both to recognize their expertise regarding development tasks and to give them an adequate organizational development mandate. When establishing an organization to coordinate infrastructure development, such as described in chapter 9, their 'shop floor IT-management' (Eriksén, 1998) and, for example, their ability to act as brokers between users and IT professionals (Kanstrup & Bertelsen, 2006) need to be recognized for the benefit of the organization.

Empirical contributions to infrastructuring from this thesis are how knowledge and decision-making about infrastructure, for good reasons, are a multidirectional and ubiquitous feature in organizations. However, as Argyris (2001) points out, although empowerment of employees is receiving a lot attention in management theory, there is a lack of examples of where empowerment to any great extent is realized in practice. The challenge of acknowledging the occurrence of a "pluralist" (Minett, 1992) management approach that extends a shop floor IT management to the organizational arena should therefore not be underestimated.

10.5.2 Conceptualizing the management of infrastructure

The action research results about organizational IT management contribute to emerging conceptualizations in related work about information infrastructures of how to manage infrastructure development. The acknowledgement of the knowledge and decision-making of shop floor users discussed in the previous section is in this sense an important start, but it is not enough. Participatory structures, processes, and tools and techniques are needed for user influence in the organizational IT management arena to work. Put in the context of Star and Bowker's (2002) paper of "how to infrastructure," the organizational IT management - put forward as a result of this PhD research - is an example of the background work of infrastructure development done in standard committees and in setting up of classifications schemes, which has been acknowledged as important but is frequently overlooked.

Two different strategies at different ends of the spectrum to manage infrastructure background were put forward by Star and Bowker (2002) – the "colonial" approach of traditional IS management of "one standard fits all" and a "democratic" model of letting "a thousand standards bloom." The outcome of this research suggests moving towards the latter end of the spectrum. At the same time, in an organizational context, Star and Bowker's (2002) conceptualization of the democratic model about letting "a thousand standards bloom," which is based on examples of large-scale infrastructure development far beyond individual organizations, comes to have too much of an open-ended connotation. As the outcome of the action research shows, when opening up infrastructure development to users on the organizational arena an un-managed "drifting" (Ciborra, 2000) infrastructure is not enough either. As discussed in the previously in this chapter in sections 10.3 and 10.4, the strategy developed with, and for, WMU builds on an organized approach: the shop floor IT management could take on increasingly complex infrastructure development through: the improved decision processes in the IT-steering committee, the new participatory and evolutionary project management, the business plans of the individual projects to coordinate development of organizational IT infrastructure, and the PD tools and techniques that support users to talk about infrastructure development.

The empirical results from this thesis thereby contribute to extend conceptualizations of democratic decision-making regarding infrastructure development in an organizational context. As described in the related work chapter, there is a revived interest in PD about organizational IT management based on user influence and democratic decision-making. The empirical results can be seen as one response to Bodker's (1996) call for new alliances between groups in organizations "with due concern for their diversity of resources, and with constructive use of the conflicts inherent in the organization." Another example that relates PD and infrastructuring is Björgvinsson, Ehn, and Hillgren's (2010) use of Mouffe's (1999) notion of "agonism" to discuss empowering a multitude of voices in democratizing innovation - which can be related inside and outside the organization (see for example Hardy and Clegg, 1996). The empirical results of this thesis show how decision-making about infrastructure is not always easy and where there can be many legitimate but sometimes diverting standpoints amongst different stakeholders that have to be mediated in a new design. Mouffe's (1999) notion of "agonism" is in this way a useful starting point for research about democratic decision-making as it extends beyond a deliberative approach in recognizing conflict as constructive controversies among adversaries who have opposing matters of concern but also accept other views as legitimate.

10.5.3 Taking a stance in shop floor IT management to manage infrastructuring

An extended development of shop floor IT management provides a path forward that is grounded in related work to the empowerment of end-users to manage infrastructuring. Shop floor IT management, which is developed in this research as driver for sustained PD in the organization, relates to the notion of Communities of

Practice (Wenger's et al. 2002)as a conceptual base (see in the related work chapter Eriksen, 1998, Dittrich et al, 2002; Karasti & Syrjänen, 2004; Karasti & Baker, 2004).

As described in the related work chapter, a yet unexplored area in both PD and infrastructuring is Wenger et al.'s (2002) extended use of the notion of Communities of Practice. Wenger et al. (2002) put forward such extensions as the "double-knit" knowledge organization and management as "cultivation." The "double-knit" knowledge organization relates to the combination of informal and formal organization, which, as previously discussed, had been a feature of the shop floor IT management at WMU even before this research started. To this end, as the empirical findings in this thesis show, shop floor IT management is not an infant stage or an auxiliary component of a comprehensive IS management scheme, but is a strategic core in managing the development of IT-support. The shop floor development constituencies at WMU can be related to not only as informal communities of practices, but also as business units with formal management responsibilities. Through the notion of "cultivation" (Wenger et al., 2002), the relation between shop floor IT management and other management in the organization is depicted. Cultivation does not rule out the managers' role in the organization in addition to shop floor IT management, but repositions their function. As shown through the empirical results at WMU, management are acting in a facilitative capacity to shop floor users' IT management through supporting structures, processes, and procedures. In other words, let managers cultivate, so that shop floor users can manage.

Shop floor IT management and its connection to organizational IT management related as a combination of informal and formal organization in the "double-knit" knowledge organization can in this way be related to the design of other frameworks of democratic organizational management. One example is Romme and Witteloostuijn's (1999) notion of the circular organization. Based on their framework, the organizational IT management approach at WMU can be discussed in terms of building on users on the shop floor being selected to represent their shop floor development constituencies in the next management level in what can be related to as connected "circles" of decision-making.

Infrastructuring in the organization will require new conceptualizations of organizational IT management. However, even though organizational IT management based on user influence is novel, this provides evidence that there are as yet unexplored paths forward in related work for future research.

10.5.4 A new role for IT-professionals

There are also implications for IT-professionals and their role in organizations. As was shown through the organizational IT management deliberations, the mandate of IT-professionals to decide about technical infrastructure was increasingly remodeled. The ability of IT-professionals to separately decide about issues pertaining to technical infrastructure was, for example, implicated by shop floor people also taking on technical infrastructure development themselves.

This does not mean that the role of IT-professionals is becoming obsolete. IT-professionals are contributing not only in playing increasingly supportive roles to end-user developers on the shop floor but also in having the expertise to contribute in the meta-design of technical and organizational infrastructure. In the context of infrastructure development, the cooperation among professional IT developers, end-user developers, and users is important in order to coordinate more substantial development by the professional developers with the EUD parts of the infrastructure. For example, the evolution of the electronic forms and contact database described in chapter 6 needed to be coordinated with the infrastructure development. The formal organization of the IT infrastructure development needed to accommodate coordination and cooperation between professional and end-user development. In this sense, the organization of the IT function at WMU, where IT-professionals are closely linked to organizational functions – even with overlapping work tasks – appears to be a model worth considering further when it comes to infrastructure developments.

This also strengthens the requirements of the IT-professionals to have socio-technical expertise and calls for new areas of expertise such as aiding end-user developers in developing new types of participatory tools and techniques that are suitable for EUD (see discussion in 10.1 and 10.4.5). As the need for an expanding technical and organizational infrastructure is shown (chapter 7 and 8), both internal and external cooperation is required between shop floor people as end-user developers and IT-professionals.

10.5.5 Evolvement of infrastructure in expanding circles of located accountability

In sustaining PD in the organization, the results of this research show how a process of different layers of infrastructure development can evolve. A challenge with reference to Pipek and Wulf's (2005) framework of infrastructural layers of technology development activities is to connect points of infrastructure that take their stance in-situ design work with infrastructural background work. The approach to target the process of infrastructure development put forward in this thesis can be related *to in terms of expanding circles of located accountability.* The particular focus is how to connect situated IT management on the shop floor to a participatory organizational IT management, based on the need to coordinate technical base development. The importance of considering infrastructure background work, such as standard setting, was also highlighted by Star and Bowker (2002); it was also recognized that this is a dimension of infrastructure development that has received comparably little attention.

As is described in chapter 7, 8, 9, all infrastructure development explicitly need to its stance in issues and opportunities in particular shop floor development constituencies. Systematically anchoring to situated work practices and real development requirements facing users, as was discussed in section 10.1 is as an important enabling condition to sustain PD in the organization. In this sense, the evolution of the infrastructure development has gradually been pushed by the people on the shop floor themselves, thus giving a "located accountability" (Suchman, 2002) of design. This approach can also be related to in terms of Suchman's (1987; 2007) “plans as resources for action,” where the intent of organizational IT management structures, processes, and representation improvements have a function based on a usefulness criteria to support the situated action on the shop floor.

One characteristic shown in the empirical research that denotes the process of infrastructure development is moving between different arenas of design: local development in shop floor development constituencies in the center and then moving back and forth between technical and organizational infrastructure dimensions. This was a practice that was initiated already in the first action research cycle showing how the technical base matters (chapter 7). The early homegrown approach of using reflection papers to support the local need for infrastructure development later paved the way for additional improvements in response to other local needs of a participatory project management, as well as PD tools and techniques (chapter 8 and 9). As has been described in the research approach chapter and throughout the empirical chapters, the evolution of the infrastructure is made of interconnected cycles of infrastructure development, which gradually expand. This is also illustrated in the overview in chapter 4.3.5 regarding how the Cooperative Method Development Cycle has been applied. In the first action research cycle there was a rapid need for expansion related to the technical base (chapter 7): from the "PD one to one" approach in regard to the development of course administration support between specialization professor Wang and an IT-professional, to the inclusion of more faculty assistants and faculty in design, and finally achieving additional synergies through integrating with shop floor development constituencies related to the external website, library, alumni, and research projects. In the second action research cycle on improving how shop floor people could participate in organizational IT management, another type of shop floor development constituencies where technical development was also not primarily dependent on IT-professionals was included (chapter 9). And finally, the application of PD tools and techniques in regard to ERP project comprehensively entailed the participation of the shop floor development constituencies (chapter 8). At the same time, it was shown in regard to the IIS plan how the shop floor people retracted their support for the evolution of infrastructure development when they perceived that it was not grounded in their working needs.

As the research process in this thesis shows, a participatory process to infrastructure development can extend from situated practices to infrastructural background work. The results at WMU show a process of an ongoing interplay between situated shop floor development practices and their evolving need for technical and organizational infrastructure support.

In regard to organizational IT management, this also sets the results of this research apart from predominant IS Management schemes. For example, compared to the implementation approach of EA (Bernard, 2005) that is based on a comprehensively defined framework, these of necessity have to be invented through a grounded approach, as documented in the empirical research in this thesis. The guidance that is offered here to sustain PD in the organization is instead positioned on a process-based level of infrastructuring about enabling the shop floor users to participate the design of their own frameworks and approaches to organizational IT management. In this way, the grounded and evolving nature that comes with a “located accountability” thus demands “non programmatic solutions” (Clement & van den Besselaar, 1993) that are situated in the work realities at hand.

10.5.6 Issue of integrated governance anchoring

The approach to sustain a participatory organizational IT management put forward in this thesis is part of an integrated governance in the organization. However, what the implications are to other areas of governance is outside the scope of the research. In the same way that IS (Bernard, 2005) needs to be integrated with other areas of governance in the organization, so also does a sustained PD approach.

Sustaining a user-centered approach in the organization is a topic receiving interest outside the PD community. Hedman and Kalling (2002), for example, discuss internal organizational structures, processes, and procedures in relation to IT and business models. The authors relate the value of bottom-up innovation conceptualized through, for example, Andreu and Ciborra’s (1996) bricolage and radical learning as emerging realization that is gaining attention.

However, as the results of this thesis also show, there is more work to be done regarding sustaining a PD approach in an organization. The final empirical contribution relates to the necessity of integrated governance considerations in the organization. This also positions the limitations of this thesis and opens up for future work. If PD is going to be sustainable as a management in the organization, PD needs to link to management areas outside the socio-technical system support in the organization.

In regard to Bernard’s (2005) areas of integrated governance, such as strategic planning, workforce planning, and program management, the outcome of the deliberations was not enough to meet all the requirements during the ERP design project, as described in chapter 8 and 9. The computer / IT-steering committee and its enhanced procedures in combination with the business plans were, for example, not sufficient to guide the process of hiring new staff or to deal with cost and return on investments calculations. It can be recognized that today there is insufficient guidance in related work to deal with such issues. This may also prompt research engagements in PD to reach outside IT and to find linkages with other research communities in regard to, for example, organizational and management sciences.

11. Conclusion

A long-term ethnographic and embedded action research study of how users can participate in the evolution of an organization's IT-infrastructures has been presented in this thesis. The objective of the action research has been to support this process by leveraging Participatory Design (PD) in organizational IT management. The empirical contributions are the result of a single case instance, but as discussed in the previous chapter and summarized here, lessons for both research and action are possible in relation to sustaining PD in organizations.

As the discussion of related work indicated: How user-oriented development can be sustained in organizations is still poorly understood. Today, predominant Information System (IS) Management conceptualizations are criticized for imposing comprehensive and standardized approaches that are not working with how situated development and innovation takes place in organizations. This is not a new critique, but it is one that is gaining in research as well as in practice in modern organizations (see for example Ciborra, 2000; Hanseth & Braa, 2001). The results of this PhD research join emerging contributions about "sustained PD" (Simonsen & Hertzum, 2013) and "infrastructuring" (Karasti & Baker, 2004; Karasti & Syrjänen 2004) to develop an alternative organizational IT management based on PD.

An opportunity presented by the empirical domain in this research was (1) an organization that already had a long-term tradition of user-centered development based on established shop floor IT management practices, but (2) that wanted help to further this approach when facing increasing needs to both technically and organizationally integrate local applications on shared infrastructure. This opened up for both ethnographic and action research opportunities of how to sustain PD in an organization. The notion of "infrastructuring" (Karasti & Baker, 2004; Karasti & Syrjänen, 2004) in relation to PD was useful in guiding the deliberations of a new type of organizational IT management. The grounded question facing the action research became how shop floor development constituencies could continuously be able to manage technical and organizational infrastructure development to support them. This required new methods and approaches. Though useful and proven approaches do exist for PD in local projects (see, for example, the MUST method by Bødker, Kensing, & Simonsen, 2004), these approaches needed to be complemented to address PD on a technical and organizational infrastructure level.

The research is set at the World Maritime University (WMU), where this researcher is employed as an IT-professional. As with any university, WMU is a knowledge-intensive organization that is dependent on front-line workers for its successful performance, whether it be faculty working with students or administrative support staff. The need for working software support includes areas such as: faculty course administration, registry academic management, as well as various administrative functions such as finance and human resources. At WMU, this software support has in many instances from the beginning of the university's operations in the 1980's been developed within the realm of different shop floor development constituencies – with or without the support of IT-professionals.

Upon employment at WMU, this researcher first worked as a dedicated resource for a shop floor development constituency that managed the development of course administration software support for the faculty. In further development of the course administration software support, however, faculty and support staff increasingly called for better socio-technical integration with other functions in the organization such as the academic management locally developed in the registry department. The same types of needs were also noticeable within and among other shop floor development constituencies. This enabled a research collaboration in the form of PhD research, with the opportunity of complementing daily work as a software developer with embedded action research.

Dittrich, Rönkkö, Eriksson, Hansson, & Lindeberg's (2008) Cooperative Method Development approach was used to guide the empirical research. In total, three interlaced research cycles combining action research and ethnographic research were developed. This resulted in empirical contributions relating the use, technology, and process for participatory IT management and evolution of IT infrastructures (Floyd, 1991):

Shop floor IT management as the driver for sustained PD: The first contribution from the empirical results reported in chapter 5 and 6 is an ethnographic account that connects to the rationale of a shop floor IT management approach in the organization. Shop floor IT management is about users conducting development together with IT-developers – either in the form of end-user developers or IT-professionals. In this way, the shop floor IT management is a hybrid between informal and formal organization, enabling a new conceptualization of management of software development in the organization. The benefit of this arrangement is that software is developed in close connection with the work activities that it supports. It is shown how a shop floor IT management approach is useful for taking advantage of the innovatory design capabilities of people working in an organization to develop useful software support. It is also shown to be important in the creation of core organizational capabilities. A redline through the empirical results is how users managing their IT development on the shop floor comes to extend from local appropriation of software to triggering and participating in infrastructure development. In addition, end-user developers provide valuable input to professional IT development activities regarding, for example, recruiting the right users to project development and prioritizing amongst development requirements

The role of the technical base in infrastructure development: The second contribution of the PhD thesis reports on how a flexible technical infrastructure is needed both for IT-professionals to support user-oriented software development and for users on the shop floor acting as end-user developers. Different dimensions, to this end, are shown through the application of the CMD framework in three cycles of action research.

A number of specific dimensions of technical infrastructure in PD became visible in the evolution of the technical base in the empirical material: the technical base enables and constrains different design alternatives of local software support. At the same time, it is shown how design of local software support with shop floor users can necessitate new infrastructure design. It is also shown how the technical base frames – support as well as constrain – the application of specific PD tools and techniques between IT-professionals and users on the shop floor. Finally, through linking separate shop floor development constituencies, it is visible how the technical base prompts management of IT development beyond local development in the organizational arena.

From an infrastructuring point of view, the technical implementation platform has to be selected and designed carefully to provide the possibility to interface with heterogeneous applications and to allow for non IT-professionals to use it for a base for End-User Development (EUD). Interfaces between EUD results and the infrastructure also indicate where coordination between professional and end-user developers is needed. In addition, in the context of the above discussion about EUD, evolving and introducing new technical platforms also impacts end-user developers taking on technical infrastructure development, and prompting their involvement in its design.

The results of the empirical research contribute to related work on "infrastructuring" through the recognition of how the characteristics of a technical base in-itself play an active role in PD design of infrastructure. If PD is going to be organizationally viable according to core principles of user influence, the technical base could thus not be black-boxed in the design deliberations with shop floor users (see, for example, Kensing, 2000)

Shop floor users managing organizational IT: The second and third application of the CMD research cycle - reported in chapter 8 and chapter 9 - is about how users on the shop floor need to participate in organizational IT management in order to relate the development of their local software support. To this end, the empirical results show how shop floor users can manage the organizational infrastructure that frames their local shop development constituencies. A number of specific improvements in regard to organizational IT management where shop floor users participate in the development of structures, processes, and representations are put forward:

- *Structuring end-user influence in the organizational arena*: working with organizational structures where end-users are influential was important in managing an increasing number of stakeholders and maintaining a participatory mode of operation in the organization also beyond the local shop floor development constituencies. It is shown how a committee-based management structure as a bottom-up management over the long-term can function in the organization.
- *Participatory and evolutionary project management:* A participatory and evolutionary project management was developed that both connected to strategic planning on the organizational IT management arena and catered for design as well as implementation. Bodker et al's. (2004) MUST method was used as a base for these two improvements: instead of one final design proposal, a evolutionary design and implementation process based on Beck and Andres' (2004) planning game with story cards was put in place; the strategic planning of local project development was connected to integrated

project planning in the organizational arena. In addition, a mechanism was connected to track the ongoing development work as the project progressed.

- *Participatory tools and techniques for infrastructuring:* Finally, it is shown how a number of common PD tools and techniques are appropriated and used in infrastructure development. If users on the shop floor are going to be influential in managing planning and capturing emergent opportunities for change of technical and organizational infrastructure, it is important that there are ways to support users to communicate about development beyond local projects. In this way, the research results show how the users could collaboratively build up their understanding of infrastructure matters.

These improvements to organizational IT management developed through the action research of this PhD answer to the challenges of managing a multitude of stakeholders and stepwise implementation processes positioned by Simonsen and Hertzum (2013) to sustain PD in the organization. Furthermore, the shop floor IT management model and working with a real situated need of infrastructure - in this case, an integrated technical base - can be seen as enabling conditions in development efforts to sustain PD in the organization.

The empirical results contribute to a new wave of PD research that is using established PD principles concerning the importance of knowledge and power diversity but in a new organizational landscape with new needs to empower a multiplicity of voices in the organization. The combined results of this thesis contribute to showing how shop floor users can manage technical and organizational infrastructure development to sustain a tarticipatory approach in an organization. This enables shop floor users to continuously develop useful software support for the benefit of both themselves and the organization in which they work. In this way, it is shown how shop floor users can be put in the center in the management of technical and organizational infrastructure development, referred to as “infrastructuring” Karasti & Baker, 2004; Karasti & Syrjänen, 2004). Based on Dittrich et al.’s (2008) CMD framework, PD has been leveraged in the organization through a research approach that can be referred to as "expanding circles of located accountability." This is a process that is denoted by an evolutionary approach of PD that takes its stance on the shop floor and incorporates technology, use, and process (Floyd, 1991), and that caters for calls to acknowledge both "use in design" and "design in use" infrastructure development (see, for example, Pipek & Wulf, 2009).

The empirical results of this thesis extend current contributions of infrastructuring by adding a planning dimension to the current focus on situated action. In this way, an alternative to predominant top-down IS management conceptualizations of infrastructure development is put forward. At the same time, there is more work to do and the empirical case instance in this thesis needs to be complemented with the results of empirical research in other organizations: acknowledging the knowledge and decision power of shop floor users on all levels of development in organizations is still novel in related work. It follows, that the same is true of the existence of purposeful management frameworks. More research is needed that re-conceptualizes both the management of shop floor users as well as the role of traditional management. Noteworthy is that issues of how sustained PD relates to integrated governance in the organization are outside the research scope of this thesis and there still are many questions left to be answered. In the context of this thesis, practical questions which linger not only as to how PD connects to strategic and financial planning, as well as where PD starts and ends in an organization, but also whether PD is of relevance only to organizational IT management or whether it is needed in other functions. The steps taken in this thesis therefore need to be accompanied by more research to make leveraging PD a commonly adoptable management option in the organization.

12. References

Ackerman, M. S., Pipek, V., & Wulf, V. (2003). *Sharing expertise: Beyond knowledge management*. USA: The MIT Press.

Agar, M. (2008). Culture: Can You Take It Anywhere? Invited Lecture Presented at the Gevirtz Graduate School of Education, University of California at Santa Barbara. *International Journal of Qualitative Methods*, *5*(2), 1.

Agar, M. H. (1996). *The professional stranger: An informal introduction to ethnography*. San Diego: Academic Press.

Andreu, R., & Ciborra, C. (1996). Core capabilities and information technology: An organizational learning approach. In B. Moingeo & A. Edmondson, *Organizational learning and competetive advantage* (pp. 139–163). London: Sage Publications Ltd

Argyris, C. (2001). Empowerment: The Emperor's New Clothes. *Harvard Business Review*, *76*(3), 98–105.

Argyris, C., & Schön, D. A. (1978). *Organizational learning: A theory of action perspective*. ADDISON-WESLEY.

Argyris, C., & Schön, D. A. (1974). *Theory in Practice. Increasing professional effectiveness*. San Francisco: Joossey-Bass.

Barn, B. S., & Clark, T. (2011). Revisiting Naur's programming as theory building for enterprise architecture modelling. *Advanced Information Systems Engineering*, 229–236.

Baskerville, R., & Myers, M. (2004). Special issue on action research in information systems: making IS research relevant to practice-foreword. *Mis Quarterly*, *28*(3), 329–336.

Bateson, Gregory. 1978. Steps to an Ecology of Mind. New York: Ballantine Books.

Beck, K., & Andres, C. (2004). *Extreme programming explained: embrace change*. ADDISON-WESLEY.

Bernard, S. A. (2005). *An Introduction To Enterprise Architecture*. USA: Authorhouse.

Bjerknes, G., Ehn, P., Kyng, M., & Nygaard, K. (1987). *Computers and democracy: A Scandinavian challenge*. USA: Avebury.

Björgvinsson, E., Ehn, P., & Hillgren, P.-A. (2010). Participatory design and "democratizing innovation" (pp. 41–50). Presented at the PDC '10: Proceedings of the 11th Biennial Participatory Design Conference, New York: ACM

Blomberg, J., & Kensing, F. (1998). Participatory design: issues and concerns. *Computer Supported Cooperative Work (CSCW)*, *7*(3-4), 167–185.

Bowker, G. (1994). Information mythology and infrastructure. *Information Acumen: the Understanding and Use of Knowledge in Modern Business*, 231–247.

Bråten, S. (1973). Model monopoly and communication: systems theoretical notes on democratization. *Acta Sociologica*, *16*(2), 98–107.

Buur, J., & Matthews, B. (2008). Participatory innovation: a research agenda. *Proceedings of the Tenth Anniversary Conference on Participatory Design 2008*, 186–189.

Bødker, K., Kensing, F., & Simonsen, J. (2004). *Participatory IT design: designing for business and workplace realities*. London: MIT Press.

Bødker, S. (2000). Coordinating technical support platforms. *Communications of the ACM*, *43*(11), 5.

Bødker, S. (1999). *Computer applications as mediators of design and use* (Vol. 28, pp. 1–98). DAIMI Report Series

Bødker, S. (1996). Creating conditions for participation: Conflicts and resources in systems development. *Human-computer interaction*, *11*(3), 215–236.

Checkland, P., & Holwell, S. (1998). Action research: its nature and validity. *System Practice and Action Research*, *11*(1), 9–21.

Christensen, M., Crabtree, A., & Damm, C. (1998). The MAD experience: Multiperspective application development in evolutionary prototyping. *ECOOP'98—Object-Oriented Programming*, 13–40.

Ciborra, C. (2000). *From control to drift: the dynamics of corporate information infastructures*. USA: Oxford University Press.

Clement, A., & van den Besselaar, P. (1993). A retrospective look at PD projects. *Communications of the ACM*, *36*(6), 29–37.

Costabile, M. F., Dittrich, Y., Fischer, G., & Piccinno, A. (2011). End-User Development - Third International Symposium, IS-EUD 2011, Torre Canne, Italy, June 7-10, 2011, Proceedings. Berlin and Heidelberg: Springer-Verlag.

Dahl, R. (1985). *A preface to economic democracy*. University of California Press.

Dahl, R. (1984). *Modern political analysis Prentice-Hall foundations of modern political sciences series*. Prentice-Hall.

Diestelkamp W, Lundberg L (2000) Performance evaluation of a generic database system. *International Journal of Computer Application* 7(3):122–129

Dittrich Y (2002) Doing empirical research in software engineering—finding a path between understanding, intervention and method development*. In: Dittrich Y, Floyd C, Klischewski R (eds) Social thinking— software practice*. MIT Press, Cambridge, pp 243–262

Dittrich, Y., Eriksén, S., & Hansson, C. (2002). PD in the Wild; Evolving practices of Design in Use (pp. 124–134). Presented at the Proceedings of the Participatory Design Conference, Malmö.

Dittrich, Y., & Lindeberg, O. (2003). Designing for changing work and business practices. *Practices. Adaptive Evolutionary Information Systems. Hershey, PA: Idea Group Inc*, 152–171.

Dittrich, Y., Rönkkö, K., Eriksson, J., Hansson, C., & Lindeberg, O. (2008). Cooperative method development. *Empirical Software Engineering*, *13*(3), 231–260.

Ehn, P., & Kyng, M. (1987). The Collective Resource Approach to Systems Design. In G. Bjerknes, P. Ehn, & M. Kyng (Eds.), *Computers and democracy: A Scandinavian challenge* (pp. 17–59). USA: Avebury.

Eriksén, S. (1998). *Knowing and the art of IT management: an inquiry into work practices in one-stop shops*. Lund University.

Eriksson, J., & Dittrich, Y. (2007). Combining Tailoring and Evolutionary Software

Development for Rapidly Changing Business Systems. *Journal of Organizational and End User Computing (JOEUC)*, 47–64.

Fischer, G. (2001). Communities of interest: Learning through the interaction of multiple knowledge systems (Vol. 2001, pp. 1–96). Presented at the Proceedings of the 24th IRIS Conference, Bergen.

Floyd, C. (1991). Software Development as Reality Construction. In C. Floyd, H. Zullighoven, R. Budde, & R. Keil--Slawik (Eds.), *Software development and reality construction* (pp. 86–100). Berlin: Springer-Verlag.

Floyd, C., Mehl, W., Reisin, F., Schmidt, G., & Wolf, G. (1989). Out of Scandinavia: Alternative approaches to software design and system development. *Human-computer interaction*, *4*(4), 253–350.

Floyd, C., Reisin, F., & Schmidt, G. (1989b). STEPS to software development with users (Vol. 89, pp. 48–64). Presented at the ESEC'89.

Floyd, I., Jones, M., Rathi, D., & Twidale, M. (2007). Web mash-ups and patchwork prototyping: User-driven technological innovation with web 2.0 and open source software. *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, 86–86.

Gantt, M., & Nardi, B. A. (1992). Gardeners and gurus: patterns of cooperation among CAD users (pp. 107–117). Presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, ACM.

Greenbaum, J., & Kyng, M. (1992). *Design at work: Cooperative design of computer systems*. Lawrence Erlbaum Associates, Incorporated.

Gärtner, J., & Wagner, I. (1996). Mapping actors and agendas: Political frameworks of systems design and participation. *Human-computer interaction*, *11*(3), 187–214.

Hanseth, O., & Braa, K. (2001). Hunting for the treasure at the end of the rainbow: standardizing corporate IT infrastructure. *Computer Supported Cooperative Work (CSCW)*, *10*(3), 261–292.

Hanseth, O., Monteiro, E., & Hatling, M. (1996). Developing information infrastructure: The tension between standardization and flexibility. *Science, Technology & Human Values*, *21*(4), 407.

Hansson, C., Dittrich, Y., Gustafsson, B., & Zarnak, S. (2006). How agile are industrial software development practices? *Journal of Systems and Software*, *79*(9), 1295–1311.

Hansson, C., Dittrich, Y., & Randall, D. (2004). Agile processes enhancing user participation for small providers of off-the-shelf software, 175–183.

Hardy, C., & Clegg, S. R. (1996). Some Dare Call It Power. In *Handbook of organization studies* (pp. 622–641). Sage.

Hedman, J., & Kalling, T. (2002). *IT and business models: Concepts and theories*. Liber ekonomi.

Henderson, A., & Kyng, M. (1992). There's no place like home: Continuing Design in Use. In J. Greenbaum & M. Kyng, *Design at work: Cooperative design of computer systems* (pp. 219–240). L. Erlbaum Associates Inc.

Hepsø, V., Monteiro, E., & Rolland, K. (2009). Ecologies of e-Infrastructures. *Journal of the Association for Information Systems*, *10*(5), 430–446.

Kahler, H. (1995). From Taylorism to Tailorability: Supporting organizations with tailorable software and... Presented at the In: Proceedings of the HCI´ 95.

Kanstrup, A. (2005). *Local Design: an inquiry into workpractices of local it-supporters*. Department of Communications. Aalborg University, Denmark.

Kanstrup, A., & Bertelsen, P. (2006). Participatory IT-support. *Proceedings of the ninth conference on Participatory design: Expanding boundaries in design-Volume 1*, 87–94.

Karasti, H. (2014). Infrastructuring in participatory design (Vol. 1, pp. 141–150). Presented at the the 13th Participatory Design Conference, ACM

Karasti, H., & Baker, K. S. (2004). Infrastructuring for the long-term: Ecological information management (p. 10 pp.). Presented at the Proceedings of the 37th Annual Hawaii International Conference on System Sciences, IEEE.

Karasti, H., & Syrjänen, A.-L. (2004). Artful infrastructuring in two cases of community PD (Vol. 1, pp. 22–30). Presented at the Proceedings of the eighth conference on Participatory design: Artful integration: interweaving media, materials and practices, ACM.

Kensing, F. (2000). Participatory Design in Commercial Settings− a conceptual framework. *Proceedings of the Participatory Design Conference*, 116–126.

Kensing, F., & Blomberg, J. (1998). Participatory design: issues and concerns. *Computer Supported Cooperative Work (CSCW)*, *7*(3-4), 167–185.

Kensing, F., & Munk-Madsen, A. (1993). PD: Structure in the Toolbox. *Communications of the ACM*, *36*(6), 78–85.

Klein, H., & Myers, M. (1999). A set of principles for conducting and evaluating interpretive field studies in information systems. *Mis Quarterly*, 67–93.

Kyng, M. (1995). Making representations work. *Communications of the ACM*, *38*(9), 46–55.

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation (Learning in doing: Social, cognitive and computational perspectives)*. Cambridge University Press.

Lehman, M. (1980). Programs, life cycles, and laws of software evolution. *Proceedings of the IEEE*, *68*(9), 1060–1076.

Lieberman, H., Paternò, F., Klann, M., & Wulf, V. (2006). *End-user development: An emerging paradigm*. Springer Netherlands.

Mathiassen L (2002) Collaborative practice research. *Information, Technology & People* 15(4):321–345

Mathiassen, L. (1998). Reflective systems development. *Scandinavian Journal of Information Systems*, *10*(1), 12.

Minett, S. (1992). *Power, politics, and participation in the firm*. USA: Avebury.

Miller, S. E. (1993). From system design to democracy. *Communications of the ACM*, *36*(6), 38.

Monteiro, E., Pollock, N., & Hanseth, O. (2012). From Artefacts to Infrastructures. *Computer Supported Cooperative Work (CSCW)*, *22*, 575–607.

Mouffe, C. (1999). Deliberative democracy or agonistic pluralism. *Social research*, *66*(3), 745–758.

Naur, P. (1985). Programming as theory building. *Microprocessing and microprogramming*, *15*(5), 253–261.

Neumann, L. J., & Star, S. L. (1996). Making infrastructure: The dream of a common language (pp. 231–240). Presented at the Proceedings of the Fourth Biennial Participatory Design Conference (PDC'96): Computer Professionals for Social Responsibility, ACM

Orlikowski, W. J. (2010). The sociomateriality of organisational life: considering technology in management research. *Cambridge Journal of Economics*, *34*(1), 125–141.

Osterwalder, A., & Pigneur, Y. (2010). *Business model generation: a handbook for visionaries, game changers, and challengers*. Wiley.

Peppard, J. (1999). Information management in the global enterprise: an organising framework. *European Journal of Information Systems*, (8), 77–94.

Pipek, V., & kahler, H. (2006). Supporting collaborative tailoring. In H. lieberman, F. Paternò, M. Klann, & V. Wulf, *End user development* (pp. 315–345). Springer Netherlands.

Pipek, V., & Wulf, V. (2009). Infrastructuring: Towards an integrated perspective on the design and use of Information technology. *Journal of the Association for Information Systems*, *10*(5), 447–473.

Ribes, D., & Finholt, T. (2009). The long now of technology infrastructure: articulating tensions in development. *Journal of the Association for Information Systems*, *10*(5), 375–398.

Romme, A., & van Witteloostuijn, A. (1999). Circular organizing and triple loop learning. *Journal of Organizational Change Management*, *12*, 439–453.

Robson, C. (2002). *Real world research: A resource for social scientists and practitioner-researchers*. Wiley-Blackwell.

Rönkkö, K., Dittrich, Y., & Randall, D. (2005). When plans do not work out: How plans are used in software development projects. *Computer Supported Cooperative Work (CSCW)*, *14*(5), 433–468.

Rönkkö, K., Hellman, M., Kilander, B., & Dittrich, Y. (2004). Personas is not applicable: local remedies interpreted in a wider context, 112–120.

Sahay, S., Monteiro, E., & Aanestad, M. (2009). Configurable Politics and Asymmetric Integration: Health e-Infrastructures in India. *Journal of the Association for Information Systems*, *10*(5), 399–414.

Schuler, D., & Namioka, A. (1993). *Participatory design: Principles and practices*. USA: Lawrence Erlbaum Associates.

Simonsen, J. (2009). The Action Research Project. *IRIS 2009*, 1–11.

Simonsen, J. (2007). Involving top management in IT projects. *Communications of the ACM*, *50*(8), 52–58.

Simonsen, J. (1999). How do we take Care of Strategic Alignment? *Scandinavian Journal of Information Systems*, *11*(1), 6.

Simonsen, J. (1997). Linking design to business strategy through functional analysis (pp. 1314–1327). Presented at the Proceedings of the 5th European Conference on Information Systems, Cork.

Simonsen, J., & Hertzum, M. (2012). Sustained Participatory Design: Extending the Iterative Approach. *Design Issues*, *28*(3), 10–21.

Simonsen, J., & Robertson, T. (Eds.). (2012). *Routledge handbook of participatory design* (pp. 1–356). Routledge.

Star, S., & Bowker, G. C. (2002). How to infrastructure. In L. A. Lievrouw & L. S, *Handbook of New Media - Social Shaping and Consequences of ICTs* (pp. 151–162). London, UK: Sage Publications Ltd.

Star, S., & Ruhleder, K. (1996). Steps toward an Ecology of Infrastructure: Design and Access for Large Information Spaces. *Information systems research*, *7*(1), 111–134.

Star, S., & Ruhleder, K. (1994). Steps towards an ecology of infrastructure: complex problems in design and access for large-scale collaborative systems. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 253–264.

Suchman, L. A. (2007). *Human-machine reconfigurations: Plans and situated actions*. Cambridge Univ Pr.

Suchman, L. (2002). Located accountabilities in technology production. *Scandinavian Journal of Information Systems*, *14*(2), 91–106.

Suchman, L. (1994a). Working relations of technology production and use. *Computer Supported Cooperative Work (CSCW)*, *2*(1), 21–39.

Suchman, L. (1994b). Do categories have politics? *Computer Supported Cooperative Work (CSCW)*, *2*(3), 177–190.

Suchman, L. A. (1987). *Plans and Situated Actions: The Problem of Human-Machine Communication (Learning in Doing: Social, Cognitive and Computational Perspectives)*. Cambridge University Press.

Ure, J., Procter, R., Lin, Y., Hartswood, M., Anderson, S., Lloyd, S., et al. (2009). The Development of Data Infrastructures for eHealth: A Socio-Technical Perspective. *Journal of the Association for Information Systems*, *10*(5), 3.

Wenger, E., McDermott, R., & Snyder, W. M. (2002). *Cultivating communities of practice: A guide to managing knowledge*. USA: Harvard Business School.

Wenger, E. (1998). *Communities of Practice — Learning, Meaning, and Identity*. Cambridge, England: Cambridge University Press.

Wessels, B., Walsh, S., & Adam, E. (2008). Mediating Voices: Commumty Articipation in the Design of E-Enabled Community Care Services. *The information society*, *24*, 30–39.

Appendix 1 - Participatory Observations, Workshops, Interviews, and Meetings

The below table outlines the participatory observations, workshops, interviews, and meetings mapped in the timeline of chapter 4. These were selected from the complete recorded empirical material for further analysis to support the account given in the research diary and project documentation.

▲ *07/06/22*, Wang **EUD M**	Interview with Vice-President (Academic) Wang to establish research focus. Conducted in the office of Wang
◆ *08/12/16*, Larsson **IT**	Inline and in-depth design analysis workshop with IT-Professional Larsson about technical platform linking of Academic Administration to external and research websites. Provided an additional perspective of the software development approach at WMU. Both Jönsson and I were employed as IT-professionals with formally the same work description, but with different competency profiles that affected the work approach of how the development software support was carried out. However, together with the shop floor people, we both realized issues regarding the technical base
◆ *08/12/18*, Levy **U**	Design meeting with Faculty Assistant Levy about course administration use and design. Follow-up to reflections in research diary about user participation in design process at WMU
● *08/12/12*, White **EUD** ▲ *08/12/19*, White **EUD**	Participatory observation and interview with White about electronic forms and contact database and their linking to Academic administration. Provided first user perspective of development of design process at WMU outside the faculty. Conducted in White's office in front of White's computer. White practically showed how she worked with the electronic forms and the contact database. At the same time, she exemplified her approach regarding the design and implementation process. For this purpose sketches and paper printouts of relevant artifacts were also used.

09/02/03, Levy, **U** 09/02/17, Hansson, **U** 09/02/17, Mercier, **EUD** 09/02/23, Schulz, **U**	Design workshops with Faculty Assistant Levy, Professor Hansson, Professor Mercier, and Schulz about academic administration development on new technical platform. In addition, input to need of improved local and organizational IT management when managing design process with increasing number of users. In addition contributed to triangulating the empirical data of the close-knit shop floor development in the first research cycle
09/02/26, Larsson, **IT**	Retrospective and Evaluation workshop with IT-Professional Larsson of technical platforms and organizational IT management. Mapping of timeline. Complemented and confirmed entries in research diary from a professional IT development perspective.
09/04/20, Wang, **EUD, M,** Levy, **U** 09/04/21, Levy, **U**	Workshop with reference to timeline with Vice-President (Academic) Wang and Faculty Assistant Levy about academic administration. Complemented and confirmed entries in research diary from a user and manager point of view. Additional interview with Faculty Assistant to get a perspective from a user perspective only.
09/04/28, White, **EUD**	Meeting with Administrative Assistant White about Electronic forms development. First meeting about me participating in development project outside the faculty. Enabled perspectives of the dynamics of another shop floor IT management approach
09/05/11, Wang, **EUD, M,** Levy, **U**	Workshop with Vice-President (Academic) Wang and Faculty Assistant Levy about new and expanded Academic Administration development, also in connection to new organizational IT management. Gave input to following meeting with IT-coordination group and IT-steering committee.
09/05/13, Bouchard, **M**, Jönsson, **IT**, Nilsson, **IT**	Design Workshop about need of new organizational Management with Professor and Chair of IT-coordination group Bouchard, IT-professional Jönsson, and IT-professional Nilsson. Presentation by me of possible participatory design approaches. Discussion about how an applied approach could be developed at WMU
09/07/07, Bouchard, **M**, Jönsson, **IT**, Larsson, **IT**	Follow-up workshop about implementation of new organizational IT management with Professor and Chair of IT-coordination group Bouchard, IT-professional Jönsson, and IT-professional Nilsson.

◆ *09/08/06*, Registry	Meeting with staff of Registry department about their development, management, and technical to academic administration.
◆ *09/12/01*, Bouchard, **M** ▲ *09/12/01*, Bouchard, **M**	Meeting and interview with Professor and Chair of IT-coordination group Bouchard to follow up on the organizational IT management changes..
▲ *09/12/16*, White, **EUD**	Yvonne Dittrich interview with Administrative Assistant White about electronic form and contact database, including relation to organizational IT management. The purpose of the interview was both to relate back to findings of the participatory observation sessions and to inquire about White's relation to official IT-development beyond the specific EUD activities. Her experience of a number of changes in the way IT development has been organized over time provided valuable input to the understanding of the impact on the EUD practices.
◆ 10/02/18, IT-Coordination ◆ 10/02/23, IT-Coordination	IT-coordination meeting about 2010 work plan with new organizational IT management.
◆ 10/03/05, IT-Steering committee	IT-steering committee meeting about 2010 work plan with new organizational IT management.
● *10/05/03*, Ocampo, **U** ● *10/05/03*, Olson, **U** ● *10/05/11*, Garcia, **U**	Participatory Observations of Finance Officer Ocampo, Human Resource Officer Olson, and Finance Assistant Garcia about their respective work functions in the context of new integrated ERP system.
6 x ● *10/05/06* – Davis, **EUD, M** *10/08/16*	Participatory observations of registry system and integrated academic administration with faculty. These were done in the context of an upcoming ERP project where the intention was to understand if and how the current registry system could be integrated in the new application environment. All sessions were carried out in Davis' office
● 10/08/20, Olsson, **U** ▲ 10/08/18, Olsson, **U**	Follow-up Participatory Observation and interview with Human Resource Officer Olson about functional design of work functions in the new integrated ERP system.
◆ *10/09/29*, ERP Agresso ◆ *10/10/06*, ERP Navision ◆ *10/11/19*, ERP OpenERP	Design meetings with ERP providers and cross-departmental stakeholders.
● *10/12/10*, Magnusson, **U**	Participatory Observation with Registry Assistant Magnusson about registry system to get an alternative

	perspective to participatory observations carried out with Davis.
10/12/21, Magnusson, **U**, Davis, **EUD**,	Interview with Registry Assistant Magnusson and Registrar Davis about integrated academic administration between registry and faculty.
11/02/14, IT-Steering committee	IT-steering committee meeting about ERP design and organizational IT management.
11/03/24, IT-Steering committee	IT-steering committee meeting about ERP design and organizational IT management.
11/03/25, IT-Coordination	IT-steering committee meeting about ERP design and organizational IT management.
11/05/04, Laine, **M**	Interview with Chair of IT-steering committee to understand how he worked with process improvements of the committee work and to understand his view of the results
11/09/28, White, **EUD**	Workshop with White about development of new integrated electronic forms.
11/10/12, IT-Coordination	IT-coordination group meeting about organizational IT management.
12/05/06, Evans, **U**, Magnusson, **U**	Retrospective and Evaluation workshop with Student Social Officer Evans and Registry Assistant Magnusson. Input to mapping of timeline.
12/05/016, Hughes, **EUD**	Interview with specialization professor Hughes about the evolution of the "native" shop floor development model at WMU
12/05/29, Evans, **U**, Magnusson, **U**	Follow-interview with Student Social Officer Evans and Registry Assistant Magnusson in reference to timeline.