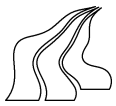
Aalborg University
Department of Computer Science, Fredrik Bajers Vej 7E, 9220 Aalborg Øst



TITLE:
Tailoring Knowledge Management Systems:
Embracing Soft Systems Thinking

SUBJECT:
Knowledge Management

PROJECT GROUP:
E3-211a

PROJECT MEMBERS:
Jens Henrik Hosbond
Rasmus Ørtoft

SUPERVISOR:
Peter Axel Nielsen

SEMESTER:
DAT6

PROJECT PERIOD:
1st of February 2003 to
4th of June 2003

NUMBER OF COPIES:
6

NUMBER OF PAGES:
142 pages

Abstract:
This Master's thesis concerns the problematic issue of tailoring knowledge management systems through application of Soft Systems Methodology (SSM) and prototyping.

The objective is to explore how the process of tailoring knowledge management systems can benefit from using SSM and prototyping.

In collaboration with WM-data, we conducted an in-depth SSM analysis followed by a series of experiments using prototypes. The prototypes were developed in order to validate the relevance of the underlying system ideas through actual use by the end-customer.

Our main contribution is a model comprising an analytical mode of operation (SSM) and an experimental mode of operation (prototyping). We have found that the two methodologies compensate each others weaknesses well and together suit the needs for tailoring knowledge management systems.

Summary

As companies and organizations around the world struggle to maintain a competitive edge, the focus on knowledge as an important internal resource and a key asset increases.

The result is that, several organizations have invested heavily in of-the-shelf knowledge management systems that do not fit the very characteristics of the particular organization. This has led to several reports on unsuccessful implementations of knowledge management systems. Schultze and Boland report success rates as low as 30% caused by technologists' lack of ability to understand the work practices of user communities [Schultze and Boland, 2000].

The rate of failure speaks for itself. We therefore set out to explore another approach for the tailoring of knowledge management systems, an approach that to a greater extent foster cooperation between end-users and systems developers. This should lead to knowledge management systems tailored to the specific needs and characteristics of the organization in question. Our research is aimed at answering the following question:

> **Research question:**
> *How can the process of tailoring knowledge management systems benefit from using Soft Systems Methodology and prototyping?*

We note that this study represents the second part of a one-year study based on improving knowledge-related work practices in a Danish sub-division of a large Scandinavian software company. Prior to this study, we have conducted an empirical investigation of current knowledge practices within the particular sub-division. The results of our prior study have been applied in the present study.

Research Approach

The process of this study has been based on an action case approach, a hybrid research approach between the action research approach and the case study research approach [Vidgen and Braa, 1997]. In order to address the very intangible

nature of knowledge-related problems, we have applied Soft Systems Methodology (SSM) as is it enables us to consider several perspectives of a problematic situation [Checkland and Scholes, 1990]. We believe that this is required as company culture, communication style, management style etc. all have an impact on how knowledge is shared and created. These issues must therefore also be considered. This also represents the main reason for choosing SSM as our tool of analysis. In addition, we need to realize the analytical findings, e.g. human activity systems from SSM. We found prototyping to be useful due to its experimental approach. By combining these two approaches, we utilize an analytical mode of operation and an experimental mode of operation in which SSM and prototyping compensate for each others weaknesses in accordance with Mathiassen and Stage's 'Principle of Limited Reduction' [Mathiassen and Stage, 1992].

Results

The results of this study relate to our experiences of applying SSM and prototyping. All our experiences will not be accounted for here. However, the combination of SSM and prototyping lead to the construction of a model representing our actual process through this study. The model serves as our main contribution and is shown in Figure 1.

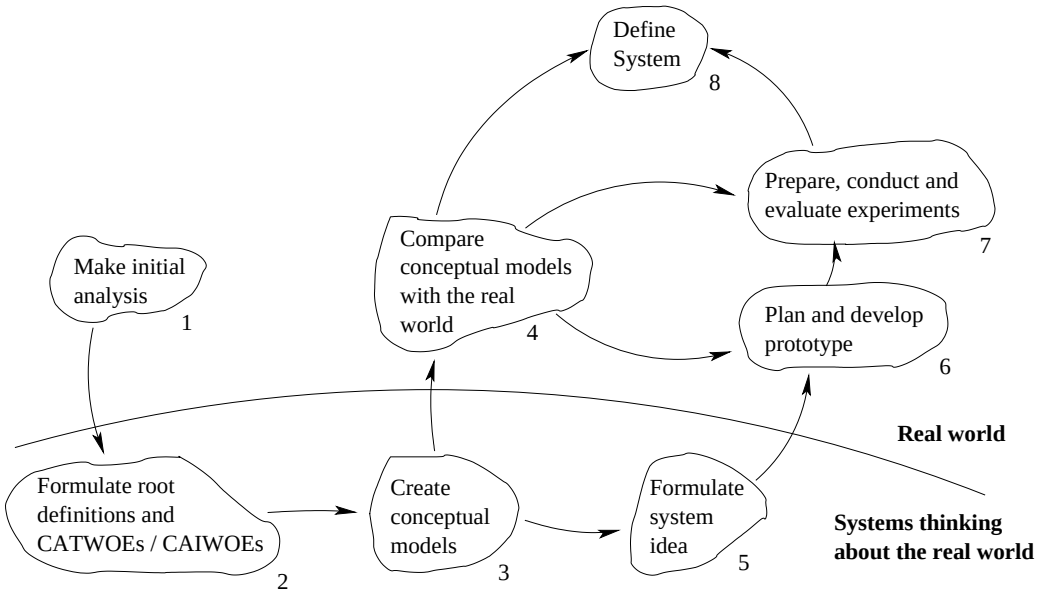


Figure 1: *Our proposed model for tailoring knowledge management systems. A combination of SSM and prototyping.*

By combining SSM and prototyping, we were able to test the validity of the analytical results from our SSM analysis by conducting experiments on prototypes

developed to support and represent the underlying system ideas of the systems from the SSM analysis. The result of the model ('Define system' - stage 8) is therefore comprised of both a knowledge management application and a representation of the knowledge activities surrounding the knowledge management application. The result of stage 8 ('Define system') can therefore be viewed as a blueprint of the required properties that needs to be in place before actually developing the tailored knowledge management system.

We found the model to be useful as we tailored two knowledge management applications for use in real-world practices.

Foreword

This report is the result of a study conducted on DAT6 and represents our Masters thesis. The study spans the period from the 1st of February to the 4th of June 2003 and has taken place at the Department of Computer Science at Aalborg University.

Prior to this study, we conducted an empirical investigation of software process knowledge in the Danish Systems Development branch of WM-data. This study lead to an increased interest as to how knowledge management systems were, to a greater extent, tailored to the characteristics of the organization. This interest has guided the research of the present study.

Acknowledgments

First, we would like to thank WM-data for a fruitful collaboration through both the present and previous research period. In particular, we would like to pay our special thanks to our contact person at WM-data, Flemming Pentz Madsen, project manager at the sub-division in Aalborg, for always showing an interest and willingness to support our study by participating in experiments, facilitating meetings and establishing contacts to other relevant groups or employees at WM-data.

Furthermore, we would like to thank our supervisor, Peter Axel Nielsen, Associate professor and Head of Department of Computer Science at Aalborg University, for an enriching collaboration in which we benefited from the highly relevant feedback and suggestions that he provided.

Jens Henrik Hosbond

Rasmus Ørtoft

Contents

Chapter 1

Introduction

Decades ago, society changed from an industrial society into an information society. In effect, this changed the characteristics of the primary raw material used in organizations. In an industrial society, capital was the primary raw material due to the requirement of large and expensive production machinery. The introduction of information technology, *IT*, opened new business markets and the most important raw material became information. Today, the information society has progressed towards a knowledge society, focusing on knowledge and creativity. Knowledge and creativity can only be extracted from one source, namely people. Therefore, there is a need for tools or methods to extract and further develop individual knowledge. One way to exploit existing knowledge and create knowledge in organizations is to foster collaboration between employees, both inside and outside the organization. However, relying only on personal contact as the primary source of sharing and creating knowledge is rather ineffective and requires a great deal of resources with respect to the time spent on this activity. Furthermore, created knowledge only exists within the minds of employees and is therefore very intangible in nature. This raises a problem with respect to accessing the knowledge of co-workers. All these problems fostered the introduction of knowledge management systems during the 1990's. Knowledge management systems should ease the sharing, capturing and access to personal knowledge in an effective way. A prime example of the early ideas on knowledge management systems is 'The Experience Factory' proposed by Basili et al. [Basili et al., 1994].

'The Experience Factory' consists of a database management system that stores knowledge, thereby making it persistent. Furthermore, a team is assembled with the primary task of maintaining the database and supplying the users with the necessary knowledge. No matter how good the idea of 'The Experience Factory' might sound, the use of it can be less successful. The basic idea of the 'The Experience Factory' suffers from the belief that a system can solve the task alone and that the organization has to adapt to the system and not vice versa. Furthermore, the team managing 'The Experience Factory' does not know what knowledge to store and maintain, resulting in a supply-driven approach in the terms of Swan et al. [Swan et al., 1999]. A team

using the supply-driven approach collects all knowledge that can prove to be useful at some point in time. For the people managing 'The Experience Factory', this means that they stand the risk of updating knowledge that would never be put to use. Instead, a more needs-based approach is required in order to focus the management of resources solely on knowledge that is requested by end-users. A needs-based approach also avoids suffering from knowledge overload which often becomes the problem of a supply-driven approach. A needs-based approach is what Swan et al. refer to as a demand-driven approach [Swan et al., 1999].

During the past decade, several organizations have invested heavily in of-the-shelf knowledge management systems that might not fit the very characteristics of the particular organization. This has led to several reports on unsuccessful implementations of knowledge management systems. Schultze and Boland report success rates as low as 30% caused by technologists' lack of ability to understand the work practices of user communities [Schultze and Boland, 2000].

The rate of failure speaks for itself. We therefore set out to explore another approach for tailoring knowledge management systems, an approach that to a greater extent foster cooperation between end-users and systems developers. This should lead to knowledge management systems which are *tailored* to the specific needs and characteristics of the organization in question.

1.1 Background

This study is based on the findings of our previous study reported by Hosbond and Ørtoft [Hosbond and Ørtoft, 2003]. Our previous study took outset in a case study at the Systems Development branch of WM-data, Denmark. The study was based on the following research question:

> *How does WM-data pursue knowledge management activities in their current software practices?*

The research question was investigated by conducting an empirical investigation of the actual software development practices at WM-data and by mapping the empirical data to related knowledge management theories. The findings of this study, however, raised several new questions inside the knowledge management arena that would require further exploration. Exploration of these questions could contribute to a greater understanding of the limits and requirements of knowledge management strategies and knowledge management systems. Furthermore, exploration could contribute to an improved notion on how tailored knowledge management strategies and knowledge management systems in the future are to be designed and implemented.

1.2 Intentions

With outset in the background of this study, Section 1.1, we are interested in exploring how we may be able to tailor knowledge management systems so that they fit the very specifics of the customer or end-user. We believe that an answer to this question will affect the probability of success, in the implementation and adaption of knowledge management systems, in a positive manner. Following this reasoning, we intend to contribute to the discussion in the knowledge management literature on how to tailor knowledge management systems. We will contribute by developing a model for tailoring knowledge management systems based on our experiences. It is our intention that the model will provide an abstract definition of the activities within the future knowledge management system, e.g. which knowledge activities are to be initiated and how? Furthermore, the process of the model must result in a prototype of a knowledge management application that is intended to guide the development of the tailored knowledge management system and to visualize activities and their interrelations for the systems developers and the end-customer. With respect to a knowledge management application, this can be compared to the database management system in the 'The Experience Factory' [Basili et al., 1994]. We emphasize that we apply the term a 'knowledge management system' in the sense that it covers both the abstract definition of the system, as mentioned above, and the knowledge management application.

The model is a result of the action case research approach used in this study. The action case proposed by Vidgen and Braa is a research approach between the action research approach and the case study research approach [Vidgen and Braa, 1997]. The action case research approach is explained in further detail in Subsection 2.2.4.

A part of our action case research approach is the development of prototypes. The applicability of these prototypes will be tested through a series of experiments and oral discussions acting as evaluation sessions. We expect that the experiments and evaluations will prove or disprove the application and relevance of the model. These intentions lead to the following research question that represents our intentions and acts as the main theme of this study.

> **Research question:**
> *How can the process of tailoring knowledge management systems benefit from using Soft Systems Methodology and prototyping?*

The question contains two aspects, namely Soft Systems Methodology (SSM, for short) and prototyping. SSM is a general problem solving tool initially introduced by Peter B. Checkland [Checkland, 1975]. The main strength of SSM lies in the ability to deal with very complex human-related problems, e.g. organizational problems related to designing and implementing knowledge management systems. SSM takes on an analytical approach to express and visualize the initial problem. Furthermore,

it offers a tool to for modelling human activity systems capable of improving the initial problem at an abstract level. The intention is therefore to take advantage of SSM when analyzing the problems related to knowledge management systems in WM-data. The success of knowledge management systems depends not only on whether or not the system satisfies the demands of the end-user, but also on whether or not the system fits the characteristics of the surrounding environment. This is however not a straightforward task to solve and the problem needs to be perceived from various angles in order to end up with an idea for a knowledge management system capable of living up to the expectations of the customer. We believe that by applying SSM we are able to explore these various perspectives and based on close collaboration with the end-user, we are able to tailor a system suited for the needs of the customer and the surrounding environment.

The second aspect of our research question concerns prototyping. Prototyping is a software development methodology that has been known for decades. It is excellent for designing and developing systems that is tailored to the specific needs of a customer or end-user. Furthermore, prototyping takes on an experimental approach to systems development compared to SSM. Taking on a development method like prototyping also calls for conducting several experiments based on the developed prototypes. This creates an effective and fast way of evaluating and adjusting the prototypes that reflects the requirements of the end-user. As argued before, the characteristics of a knowledge management system must fit the requirements for the end-user in order to be successful. Therefore, prototyping is interesting in the development of tailored knowledge management systems.

Our reason for choosing a combination of an analytical approach (SSM) with a more experimental oriented approach towards systems development (prototyping) is due to our experiences and perceptions of actual software practices at WM-data and through the dilemmas of knowledge management systems development introduced in the knowledge management literature. Another justification for the combination of an analytical approach with an experimental approach is suggested by Mathiassen and Stage [Mathiassen and Stage, 1992]. They reflect on different methods of designing and developing computer-based systems. They propose that in order to reach a rational and successful approach to developing software systems, software development must partly be based on an analytical mode of operation and partly on an experimental mode of operation in order to address each others main weaknesses: Uncertainty and complexity, respectively. An analytical mode of operation is conducted through a number of steps in which the initial problem is decomposed into smaller sub-problems with less complexity. An example of a model adopting an analytical mode of operation is the well-known waterfall model. The ability to reduce complexity is one of the main strengths of an analytical mode of operation. It is required that information about the properties of the customer is present. But how much information is necessary? This also introduces the primary weakness of this mode of operation, namely that an analytical mode of operation will result in a simplified understanding and representation of reality, thereby adding uncertainty

to the mode of operation.

To complement this weakness of an analytical mode of operation, Mathiassen and Stage propose the usage of an experimental mode of operation [Mathiassen and Stage, 1992], e.g. prototyping. The main strength lies in the conduction of experiments in which design ideas and functionality are tried out in practice and evaluated with the purpose of improving the prototypes. Experimentation and evaluation reveal the characteristics of the organizational setting, reflecting the end-user's environment and implying a reduction of the uncertainty regarding the initial problematic situation. An example of a model belonging to an experimental mode of operation is the evolutionary development model [Floyd, 1984]. However, the experimental mode of operation does not present ways of reducing complexity.

Based on the reasoning described above, Mathiassen and Stage propose 'The Principle of Limited Reduction' stating the following [Mathiassen and Stage, 1992, pp. 173]:

> **The Principle of Limited Reduction:**
> *Relying on an analytical mode of operation to reduce complexity introduces new sources of uncertainty requiring experimental countermeasures. Correspondingly, relying on an experimental mode of operation to reduce uncertainty introduces new sources of complexity requiring analytical countermeasures.*

The discussion and the above proposition of Mathiassen and Stage supports our belief and perception of what a model for tailoring knowledge management systems must possess. Furthermore, it justifies the combination of SSM and prototyping.

Through our study, we gain a thorough understanding on how to combine the analytical method of SSM with the more experimental approach of systems development of prototyping. We form a model for the tailoring of knowledge management systems based on both methodologies. The model is a result of our applications of both SSM and prototyping. Through our work, we therefore intend to contribute to a discussion on how SSM and prototyping together fit the tailoring requirements for knowledge management systems. Furthermore, we intend to contribute with tailored prototypes capable of improving the problematic areas of P^+. As argued previously, the purpose of P^+ is to improve the software development practices through sharing of project management experiences. Therefore, they are interesting from a knowledge management perspective.

1.3 Outline

This report starts off by describing the research approach that we have applied throughout this study, Chapter 2. We present the action case together with the

theory behind SSM and prototyping which represent our means of analysis and experimentation. In the research setting, Chapter 3, we present a short description of the knowledge management theory that is relevant to this study. Furthermore, we shortly present the organizational setting of WM-data and of P^+ in particular. Based on our result from our previous study [Hosbond and Ørtoft, 2003] and our cooperation with WM-data during the present research period, we present our SSM analysis in Chapter 4. In Chapter 5, we present our prototyping experiences based on two experiments. Chapter 6 presents the results of our work and is based on the experiences from the SSM analysis and the findings from prototyping. We present these findings and relate these to actual experiences and to relevant knowledge management theory presented in Chapter 3. We summarize our experiences and contributions in the conclusion of Chapter 7.

Chapter 2

Research Approach

In this chapter, we present our research approach along with the two methodologies applied in our research: Soft Systems Methodology (SSM) and prototyping. Our overall research approach lies within action research. However, as it will be argued in Subsection 2.2.3, our action taking and the change imposed on WM-data is not quite substantial enough to fully constitute action research. Therefore, we address the concept of *action case* as a hybrid research approach between action research and case studies, described by Vidgen and Braa [Vidgen and Braa, 1997].

We start off by briefly reviewing qualitative research, presenting an IS research framework proposed by Braa and Vidgen [Braa and Vidgen, 1997]. Next, we present aspects of action research that are relevant to us, including its historical background. Next, we discuss limitations in our research approach in relation to specific aspects of action research. This sets the stage for introducing the action case approach. It is argued why our research approach is an action case rather than a full-scaled action research project. Furthermore, limitations of the action case are discussed.

After describing the characteristics of the action research and action case approaches, we present the two main methodologies of our research: SSM and prototyping. As argued in Chapter 1, knowledge management systems have to be tailored to fit the complex characteristics and requirements of each organizations in order to be successful. Furthermore, the problematic situations that knowledge management systems are supposed to support, often involve groups of employees with various interests. This results in very complex human-related problems. In order to understand such human-related problems, we need a tool that in a holistic way can visualize the problem in order to create a shared conception of the problematic situation. Furthermore, the method of analysis should involve the activity of suggesting several possible solutions to the problem, each representing a particular perspective on the initial problematic situation. The capabilities of SSM fit these requirements well.

Furthermore, according to Mathiassen and Stage's 'Principle of Limited Reduction' described in the intentions part of the introduction, Section 1.2, an analytical ap-

proach needs to be balanced with an experimental approach in order to address the uncertainties introduced through analysis. Prototyping is one such experimental approach that fits this purpose well.

Both SSM and prototyping is therefore addressed, but first, we turn to qualitative research, in particular the action research approach.

2.1 Qualitative IS Research

Qualitative research is often contrasted with quantitative research. A *quantitative* approach assumes that there exists an objective truth to be studied through scientific methods of systematic and statistical measurement [Creswell, 1994]. The focus is on causal determination, prediction and generalization of findings. In contrast, a *qualitative* approach presupposes reality as subjective and multiple as seen by participants [Creswell, 1994]. The focus is on understanding and often involves studying social an cultural phenomena.

Baskerville and Wood-Harper note that qualitative IS research approaches typically are illustrated with three main types of approaches: Case study research approaches, ethnographic research approaches and action research approaches [Baskerville and Wood-Harper, 1998]. Braa and Vidgen also work with the same three types of research approaches, but use a slightly different terminology: Case studies, field experiments and action research [Braa and Vidgen, 1997]. They propose an IS research framework for what they call 'the Organizational Laboratory' based on these three types of research approaches. It is argued, among other things, that "the primary laboratory for IS research is the organization, where the development and use of technical artefacts can be studied in-context and the resulting findings used to inform both the practice and theory of IS" [Vidgen and Braa, 1997, pp. 525].

Their research framework, depicted in Figure 2.1, is based on three perspectives: Positivism, interpretivism, and intervention.

Positivism has its origin in the mechanistic world view [Dahlbom and Mathiassen, 1995]. In a broad sense, positivism is based on the five ideas of objective observation, explanation and reduction, general knowledge, hypothesis testing and physicalism [Dahlbom and Mathiassen, 1995, chap. 10]. Rigor is also assumed to be a part of positivism as repeatability, reductionism and refutability are mentioned as key virtues [Checkland, 1981] and [Vidgen and Braa, 1997]. Particularly, reduction is accentuated in the research framework. In contrast, interpretivism is similar, by many standards, to hermeneutics and has grown out of the romantic world view [Dahlbom and Mathiassen, 1995]. Hermeneutics is based on the six ideas of interpretation, subjective understanding, participation, the uniqueness of the situation, dialectics and the hermeneutic circle

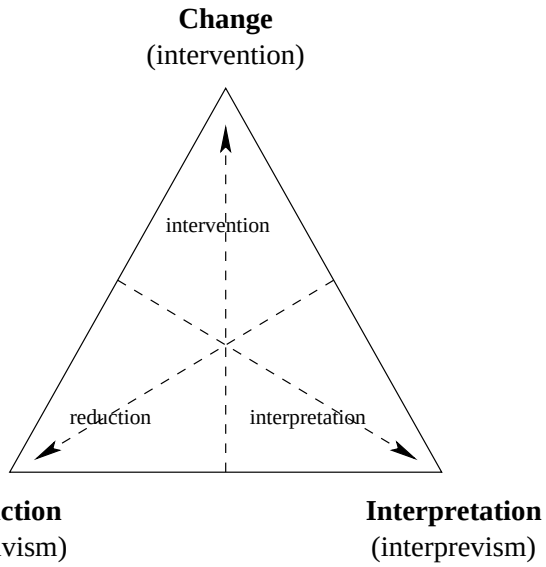


Figure 2.1: *The IS research framework proposed by Braa and Vidgen [Braa and Vidgen, 1997]*

[Dahlbom and Mathiassen, 1995, chap. 10]. Interpretivism therefore focuses on the individual and its (often contradicting) interpretations of the world. Generalized approaches are thus problematic; they do not take into account the individual differences that comprise an IS research setting [Vidgen and Braa, 1997], including social behaviour and organizational context. Intervention takes on a more radical approach to research. Here, knowledge is gained through making deliberate interventions in order to achieve desirable change within the organizational setting [Vidgen and Braa, 1997].

The three perspectives on IS research (positivism, interpretivism and intervention) each describes the underlying ideology of their respective research approaches: Field experiments, case studies and action research. In this way, these three research approaches are categorized according to the IS research framework as depicted in Figure 2.2. Field experiments thus focus on prediction, case studies on interpretation and understanding and action research on change.

After having looked briefly at qualitative IS research approaches, we now focus on the research approach that guides our research: Action research.

2.2 Action Research

Action research distinguishes itself from other qualitative IS research approaches by focusing on change. That is, change is introduced in the form of some sort of

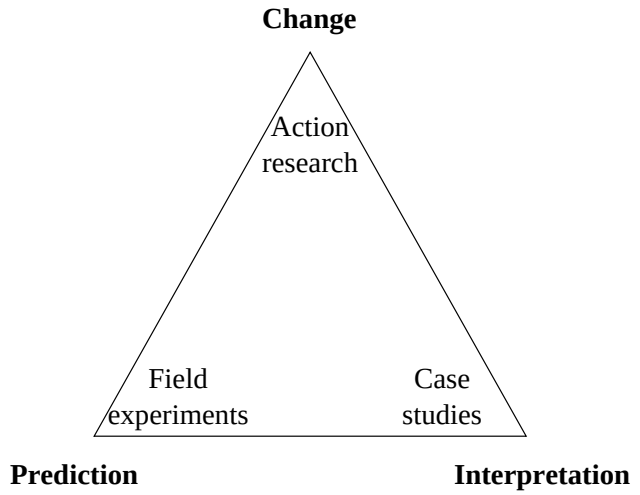


Figure 2.2: *The three main research approaches mapped to the IS research framework of [Braa and Vidgen, 1997].*

action-taking involving the people affected by the change. The aim is to study the effects and extract knowledge from the interaction between theory and practice. The main focus is on intervention; theory is tested in practice and from such intervention, new knowledge is gained about both applied theories and studied practices. Researchers typically cooperate with people from the application domain, acting as change agents, facilitators or interventionists in the process of introducing such change. One key idea is that practice cannot be understood just by observation. The researchers must be allowed to intervene into the research setting in order to (fully) understand its complex nature, including social behaviours and tacit work routines. The involvement of people from the application domain is a requisite part of action research. This rationale simply constitutes common sense. As Blum notes: "If the situation investigated is eventually to be changed, people must become involved in the research rather than being passive subjects" [Blum, 1955, pp. 2]. In this way, commitment is established towards the change so that people do not feel pressured or overlooked and accordingly generate resistance as changes are carried out.

2.2.1 Origin of Action Research

IS action research is a qualitative research approach originally adopted from social sciences. The concept of action research dates back to the publication of Kurt Lewin from 1947, [Lewin, 1947], who lead the Research Center for Group Dynamics at the University of Michigan at the time [Baskerville and Wood-Harper, 1998]. At the research center, researchers studied social psychology within the framework of field theory [Baskerville and Wood-Harper, 1998]. One of the results was the concept of

action research which they applied in their research.

The original action research model proposed by Lewin in 1947 included six stages [Baskerville and Wood-Harper, 1998]:

1. Analysis
2. Fact finding
3. Conceptualization
4. Planning
5. Implementation of action
6. Evaluation

However, the model was meant for studies in social sciences. In 1955, Blum took a broader view on action research and addressed a more general scientific arena in his view on action research as a "diagnosis of social problems with a view of helping the situation" [Blum, 1955, pp. 1]. This means that action research focuses on social aspects of the research setting in order to identify and relieve problematic situations. According to Blum, all action research therefore consist of two overall stages [Blum, 1955]:

1. A *diagnostic* stage where the problem is analyzed and hypotheses are developed.
2. A *therapeutic* stage where "the hypotheses are tested by a consciously directed change experiment, preferably in a social "life" situation" [Blum, 1955, pp. 1].

Action research has since then evolved into numerous forms which each has its own distinguishing characteristics. For more details about this evolution, we refer to Baskerville and Wood-Harper's article on diversity in IS action research approaches [Baskerville and Wood-Harper, 1998].

2.2.2 Action Research Defined

Various definitions of action research abound. The rather slim definition provided above by Blum may seem rather unprecise. Therefore, we adopt the more detailed definition of action research provided by Hult and Lenning [Hult and Lenning, 1980]:

> "***Action research*** *simultaneously assists in practical problem-solving and expands scientific knowledge, as well as enhances the competencies of the respective actors, being performed collaboratively in an immediate*

> *situation using data feed back in a cyclical process aiming at an increased understanding of a given social situation, primarily applicable for the understanding of change processes in social systems and undertaken within a mutually acceptable ethical framework*"

This definition is directly based on Lewin's six stages of action research [Baskerville and Wood-Harper, 1998]. It stresses the correlation between theory and practice in a reactive process of stimulus-response [Baskerville and Wood-Harper, 1998].

According to Baskerville and Wood-Harper, action research is characterized by the following [Baskerville and Wood-Harper, 1998, pp. 92]:

1. Its multivariate social setting
2. Its highly interpretive assumptions about observation
3. Intervention by the researcher
4. Participatory observation
5. The study of change in the social setting

All of these characteristics enable an action researcher to study complex social settings through participatory observation in the process of intervention. The interpretive assumptions about such observations relate to the fact that the results only apply to the particular social setting.

2.2.3 Why not Action Research

Although we adopt SSM and prototyping in our research as stated in the introduction, Chapter 1, it may be argued that the process of our research does not satisfy the essential requirements of intervention and change enough to constitute action research. Although SSM is classified as an action research methodology [Baskerville and Wood-Harper, 1998] because of its intervening stages (see Section 2.3 for more details of SSM), our involvement of WM-data does not fully constitute intervention, but rather we interpret our experiences at WM-data. On the other hand, our research approach does involve key individuals at WM-data in order to test theories-in-action and provide a *proof-of-concept* of our system ideas. To some extent, this may constitute intervention because our cooperation hopefully do affect WM-data in some ways. In this way, we rather combine both interpretation and intervention, using neither in their purest form. Vidgen and Braa propose a hybrid between action research and case study termed *action case*, signifying the mix between change and interpretation [Vidgen and Braa, 1997]. They use their own IS research framework of [Braa and Vidgen, 1997], described previously in this section,

to identify three such hybrids between the three main qualitative IS research approaches. This is illustrated in Figure 2.3. The hybrids are quasi experiments, hard case, and action case. Thus, our research resembles the action case rather than action research.

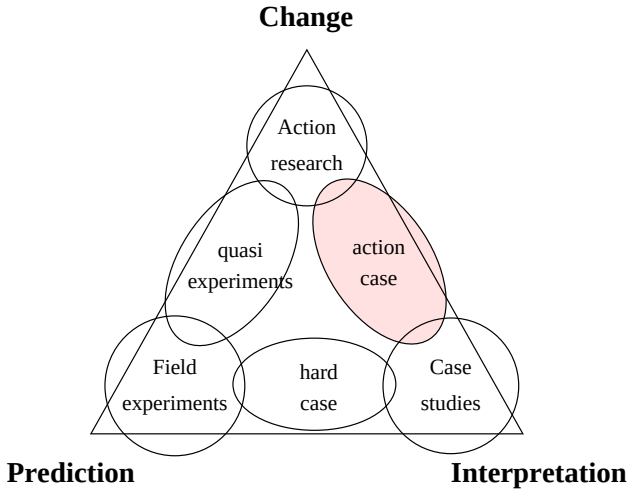


Figure 2.3: *Three hybrid research approaches adapted from [Vidgen and Braa, 1997].*

However, by using a hybrid research approach such as the action case, some problematic aspects arise that we need to be aware of.

2.2.4 Characteristics of the Action Case Research Approach

Vidgen and Braa use McGrath's notion of dilemmatics [McGrath, 1982] to identify problematic aspects of the action case approach [Vidgen and Braa, 1997]. McGrath regards the research process "not as a set of problems to be *solved*, but rather as a set of dilemmas to be *lived with*"[McGrath, 1982, pp. 69]. He uses the concept of dilemmatics to characterize the three aims of experimental research:

1. *Generalizability* with respect to populations
2. *Control* of variables
3. Existential *realism*

According to McGrath, one can maximize one of the above aims at the expense of the two others. For instance, if *realism* is maximized the researcher must downgrade *generalizability* and *control* over the research setting because realism makes the situation somewhat unpredictable. Alternatively, one might try to focus on two of the above aims, totally blocking out the possibility of achieving the third. Thus,

increasing the focus on one of the aims results in a decrease of one or both of the others.

In respect to Braa and Vidgen's IS research framework, McGrath's dilemmatics is used to argue that it is impossible for research to benefit from all of the advantages of the three purified forms of IS research (field experiment, soft case study, and action research) at the same time [Vidgen and Braa, 1997]. The following example is given: "It is not possible for a researcher to be involved with IS practice as though she/he was entirely and indistinguishably part of the organization, while also being an outsider who can stand back from the situation and make interpretations, and at the same time produce rigorous results in the positivist tradition" [Vidgen and Braa, 1997, pp. 529]. The hybrid research approaches thus make a compromise between two corners of the triangle while fully blocking the third corner. For the action case, this compromise is between interpretation and change. The compromise is made at the expense of predictability. The compromise is, however, not easy. In case studies, the researcher attempts to become an objective observer who makes interpretations based on what is being studied. This means that the researcher must be careful not to impose viewpoints and beliefs onto the research setting. Optimally, subjects in the research setting must not be aware that they are being studied because they might change behaviour. Contrary, action research aims at introducing change to the research setting and studying the effects. So what may be done to balance these two opposites? Vidgen and Braa note that researchers involved in case studies may undeliberately contribute to change just by asking questions about current practices and behaviours [Vidgen and Braa, 1997]. Also, small-scaled action research projects with a deep contextual understanding are often more appropriate than large-scaled interventions [Vidgen and Braa, 1997]. The reason is that organizations might not be willing to support such projects because of their costs. Furthermore, the topic to be investigated by the project might be better suited for a smaller-scaled effort.

2.2.5 Further Limitations of Our Research Approach

In our involvement with WM-data, the fact that we are students limits our chances of affecting changes. Large organizational changes such as changing the organizational setting of P^+ would probably be unrealistic. However, small changes such as affecting activities of P^+ would be within the our reach. In this respect, we can make recommendations based on our research and suggest changes, but the choice of whether or not these recommendations and suggestions are to be followed is up to WM-data.

We now turn to SSM which is going to be used in the process of tailoring knowledge management systems.

2.3 Soft Systems Methodology

Soft Systems Methodology, SSM for short, is a general problem solving tool that helps structure thinking about complex human-related problems. SSM has its origin in the work of Peter Checkland and colleagues from the University of Lancaster around the 1970s [Checkland and Scholes, 1990]. SSM is based on systems theory and address real-world problematic situations [Checkland and Scholes, 1990]. Through the process of SSM, potential systems are modelled that are capable of improving problematic situations. A real-world situation can be improved through various potential systems depending on the observer's beliefs and perception of the problem. Therefore, there will always be more than one system capable of improving the situation. In SSM, different perceptions (or worldviews) of a real-world situation is referred to as different *weltanschauungen*. When modelling real-world situations, SSM prescribes systems thinking in terms of abstract wholes also called holons. Systems thinking is based on the two core concepts:

- Emergence and hierarchy
- Communication and control

With respect to emergence and hierarchy, Checkland and Scholes argue that a system may have properties that refer to the whole, but are meaningless in terms of the parts that make up the whole [Checkland and Scholes, 1990]. In short, this is what Checkland and Scholes refer to as emergent properties. Furthermore, the concept of emergent properties is strongly related to the division of a system into a hierarchy. Take for example a biological organism. The organism can be considered as a hierarchy of layers constituting the whole, e.g. atoms, molecules, cells and organs just to mention a few. Each layer is described by a set of emergent properties which affects the properties of the next hierarchical level.

The second concept, communication and control, relates to the adaptive nature of the whole. The adaptive nature of wholes is related to the science of biology [Checkland, 1981]. Checkland argues that systems thinking is very much related to the science of biology. By this, he means that the unrestricted nature of biology is related to the unrestricted nature of wholes. That is, a whole is considered unrestricted and open, meaning that the whole can adapt to a continuously changing environment. A parallel can be drawn to organisms which also are able to adapt to a changing environment. For the whole to be able to adapt to these changes, requires a change of the overall properties of the whole, but also of the properties of each hierarchical layer. What is needed is communication between the hierarchical levels of the whole, informing each layer of the environmental changes. In response, each layer has to react to these changes and has to be regulated by a control mechanism, ensuring that the whole adapts to these changes.

Systems as adaptive wholes reflect systems that are dependent on human actors. In SSM, these systems (or holons) are therefore conceived of as *human activity systems*. A human activity system consists of a set of activities which are interrelated and together form a holon. In the following elaboration on SSM, we will use the terms systems, human activity systems and holons interchangeably, following Checkland and Scholes [Checkland and Scholes, 1990].

2.3.1 SSM & Action Case

With respect to the methodological characteristics of SSM, it prescribes an cooperative analysis style. That is, the main reason for adopting SSM as a primary means of analysis is when wanting to obtain a close interaction between analysts (problem solvers) and customers (problem owners) with the purpose of suggesting several systems for improving a problematic situation. Through the close interaction, problem solvers affect the process by suggesting different systems and by initiating discussions concerning the problematic situation. By doing this, the problem solvers become interventionists in some respect. However, this process does not necessarily constitute intervention, but may also be used within an action case approach, focusing more on the analytical aspects of SSM. Consequently, the characteristics of SSM fits well with an action case approach.

In the following, we present the two main parts of SSM, namely that of the logic-based enquiry and the cultural enquiry.

2.3.2 Two Streams of Enquiry

For users of SSM (problem solvers), two streams of analytical enquiry exist, namely the stream of logic-based enquiry and the stream of cultural enquiry. The stream of logic-based enquiry covers the modelling and formulation of potential systems solving or improving the real-world situation. However, in order to model relevant systems, background knowledge of the organizational context and the real-world situation is required. In SSM terms, this is referred to as the stream of cultural enquiry. The following subsections describe these two streams in sequence, starting off with the logic-based stream of analytical enquiry.

2.3.3 Logic-based Enquiry

SSM is today most often thought of as the process of Checkland's *seven stage model* depicted in Figure 2.4.

The model represents the early process of SSM, originally proposed by Peter Checkland in 1975. Given an initial problematic situation in stage one, soft systems thinkers use the seven stage model to formalize and visualize several systems to improve the

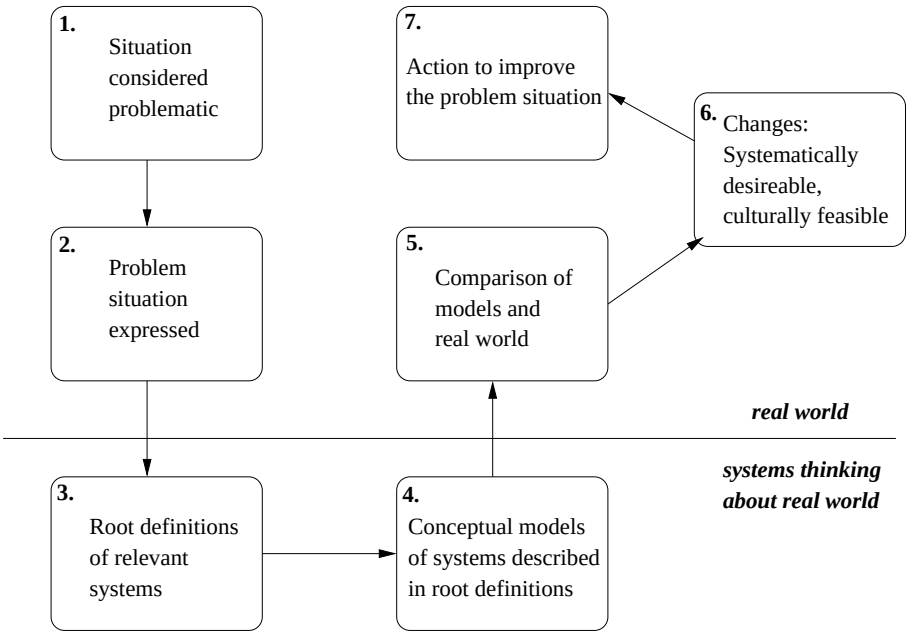


Figure 2.4: The seven stage model representing the SSM [Checkland, 1975].

situation considered problematic. That is, the proposed systems represent conceptions of how the real-world situation can be improved in terms of relevant systems. The seven stage model represents a logic-based thinking that includes selecting, formalizing, modelling and comparing human activity systems with perceptions of the real world. It should be noticed that the seven stage model is divided into two parts, namely the 'real world' and 'systems thinking about the real world'. Stages in the 'real world' represent activities in which the real-world situation is either expressed or compared via relevant systems.

A confusing element of the seven stage model in Figure 2.4 is the sequential ordering of stages. It might seem logical to simply follow the flow of the stages. This is, however, not the case. The seven stage model and SSM in general is highly iterative, and the particular sequence in which the phases of the seven stage model is conducted is not of vital importance. Depending on the situation, some stages may be repeated or even left out. Conducting only one single iteration of the seven stage model will result in narrow-minded systems, only representing a subset of the potentially relevant systems that could improve the problematic situation. In terms of systems thinking, this is similar to hard systems thinking in which analysts seek to map real-world problems into one single true system that represents the real-world situation. In contrast, soft systems thinkers, i.e. users of SSM, believe in the existence of multiple perceptions of real-world situations depending on the interests and personal beliefs to be prioritized.

Each stage of the seven stage model proposes a set of actions that reflect a part of the SSM process. We will now elaborate on the stages of the seven stage model and their proposed actions.

Stage One and Two

In everyday-life, we all experience situations that we believe could have been handled differently. We experience situations at work, at home or anywhere we go that we perceive as problematic and improvable [Checkland and Scholes, 1990]. It is such situations that are representatives for the problematic situations conceived of in stage one of the seven stage model. These situations lay the grounds for the following SSM analysis. Having identified a problematic situation, e.g. a work-related problematic situation that repeatedly arises, a way is needed to express the problematic situation in order to obtain consensus of the problem between the following roles: The problem solvers, the clients and the problem owners. These roles are further described in Subsection 2.3.4. Rich pictures are one useful way of expressing a problematic situation. A formal and linguistic description of all the aspects of a problematic situation would turn out quite cumbersome and incomplete. Most people are familiar with the saying that "a picture tells a thousand words" which further justifies the use of rich pictures. The expressive power of rich pictures is only limited by the creative skills of the SSM users. However, some rules of thumb exist on what a rich picture should consist of. A rich picture should contain structures, processes (activities), interrelations between processes, conflicts, roles (actors), norms and factual data. Structures can be in the form of databases, archives, conference rooms or other sorts of physical arrangements present in the problematic situation. When creating rich pictures, the purpose is to express all meaningful aspects of the problematic situation. However, it is worth noticing that Checkland and Scholes argue that the process of creating rich pictures is an iterative process which can be revisited during all steps of the SSM process [Checkland and Scholes, 1990].

Stage Three and Four

The rich pictures in stage two represent the problematic situation that is to be improved. To improve the situation, several perceptions of human activity systems — holons — are proposed and selected. The selected holons can accordingly be categorized as either primary task systems or issue-based systems [Checkland and Scholes, 1990]. Checkland and Scholes argue that there is no clear-cut distinction between the two categories, but rather, they are complementary extremes. Primary task systems represent systems that map to institutionalized arrangements of an organization. That is, tasks for which the organization was created. Issue-based systems, on the other hand, are relevant to mental processes that are not formally embodied in the organization and therefore often are subject of dispute and concern.

Having selected several relevant systems, one may continue with formalizing the systems through root definitions and CATWOEs. First, the intention is to formalize and name the selected systems by describing them through root definitions. A root definition is a textual definition of a system, consisting of three parts, namely *what* a system is supposed to do, *how* this is done, and *why* this is done. One way of structuring root definitions is through the use of **X**, **Y** and **Z** [Checkland and Scholes, 1990]. These annotations are related to the form of the root definition in the following way:

A system to do **X**, by means of **Y**, in order to achieve **Z**

A root definition is therefore a textual definition of a conceptual model. To help the formulation of root definitions, a CATWOE can be defined. A well-defined root definition contains each element from the CATWOE [Checkland and Scholes, 1990] However, they may also be used to supplement each other. A CATWOE express the transformation process that takes place when an input is transformed into an output by the system. A CATWOE consists of the following parts:

- **C**ustomers
- **A**ctors
- **T**ransformation process
- **W**eltanschauung
- **O**wners
- **E**nviromental constraints

Customers represent the victims or beneficiaries of the transformation process. *Actors* are the ones who conduct the transformation. The *transformation process* outlines the input and what is expected as output. In the traditional sense of SSM described by Checkland [Checkland, 1975], a transformation process takes a well-defined input and transforms it into a well-defined output without focusing on the intermediate states of the transformation process. Mathiassen and Nielsen propose another type of system, an *interaction* system, as an alternative to the transformation system of traditional SSM [Mathiassen and Nielsen, 1998]. Interaction systems may be used when the transformation process does not have a well-defined input and output. This is further elaborated on in Subsection 2.3.5.

Weltanschauung represents how the problematic situation is perceived. Different actors are most likely to have different ways of viewing and solving a problematic situation. That is, various *Weltanschauungen* exist for solving the same problematic situation. The *owners* are the people with the power to stop the transformation process and the *environmental constraints* are elements outside the system that the

system takes as given. Daily work routines in a company could affect a system and in that respect act as an environmental constraint.

In stage four, the aim is to visualize and model activities of the human activity system and how they interrelate. The outcome is a conceptual model of a potential system selected to improve the problematic situation. The activities are phrased in a verb- noun way and placed in drawn bubbles. The dependency between activities are shown by the use of arrows. An arrow going from activity A to the activity B means that A takes place before B or that B is dependent on A.

To be able to judge whether a transformation in a potential system is unsuccessful or successful Checkland and Scholes introduce the following three terms [Checkland and Scholes, 1990]:

1. Efficacy
2. Efficiency
3. Effectiveness

The first account, *efficacy*, relates to whether or not the means — Y in the roof definition — of the selected system works in producing the expected output of the transformation. *Efficiency* concerns whether the transformation is carried out with a minimum use of resources. However, a transformation that works and uses a minimum set of resources might still not satisfy the longer term goals of the system. The last account, *effectiveness*, is related to monitoring this issue. The three Es, as these accounts are called, are to be used during the comparison stage.

Stage Five, Six and Seven

Stages three and four together formed the process of systems thinking about the real-world. Activities such as selecting, formalizing and modelling potential systems that are relevant to the initial problematic situation resembles these stages. The next step is to compare the modelled systems to the real-world situation. This is done in stage five of the seven stage model. The purpose of the comparison is to locate differences between the modelled systems and the real-world situation in order to foster discussion of relevant changes to the system. A comparison can be done in several ways. Checkland and Scholes describe four methods for conducting a comparison [Checkland and Scholes, 1990]:

1. Informal discussion
2. Formal questioning
3. Scenario-writing based on the models

4. Try model the real-world situation in the same structure as the conceptual model

Of these four methods of comparison, the second method is most often used and includes participation of the users of SSM and owners of the problematic situation. Formal questioning can be conducted either as a formal interview, a discussion or repeated dialogues over time. When conducting a comparison, it is useful to formalize a spreadsheet that contains each activity from the conceptual model and the formalized questions needed in order to map the differences between the conceptual model and the real-world situation.

The outcome of the comparison is a proposal for relevant changes to the proposed systems. The discussion and selection of changes are performed in stage six.

In stage seven the changes agreed on in stage six are carried out in order to change the proposed system so that they, to a greater extent, fit the characteristics of the real-world situation.

2.3.4 Cultural Enquiry

The logic-based enquiry represented by the seven stage model, Figure 2.4, is probably the most well-known representation of the process of SSM. However, as Checkland and Scholes argue, the model lacks issues concerning the organizational context in which the problematic situation takes place [Checkland and Scholes, 1990]. In order to be successful in formalizing and modelling potential systems intended to improve the situation, users of SSM need knowledge on issues such as the *social system*, *political system* and *roles* involved in the intervention, e.g. problem solvers and problem owners [Checkland and Scholes, 1990]. These issue form the cultural enquiry. To fully exploit cultural enquiry, it must be possible to access knowledge on the organizational context at all times throughout the process of SSM. The cultural enquiry and the logic-based enquiry form a new process of SSM compared to the seven stage model. It is believed to be a better representation of the actual process of SSM due to the presence of the cultural enquiry. The combination of cultural enquiry and logic-based enquiry yields the overall process of SSM shown in Figure 2.5.

Checkland and Scholes explain the cultural enquiry by dividing it into three parts: Analysis One, Analysis Two and Analysis Three [Checkland and Scholes, 1990]. Analysis One relates to the roles of the intervention itself, Analysis Two represents the social system and Analysis Three relates to the political system of the problematic situation. The three types of analyses are now further explained.

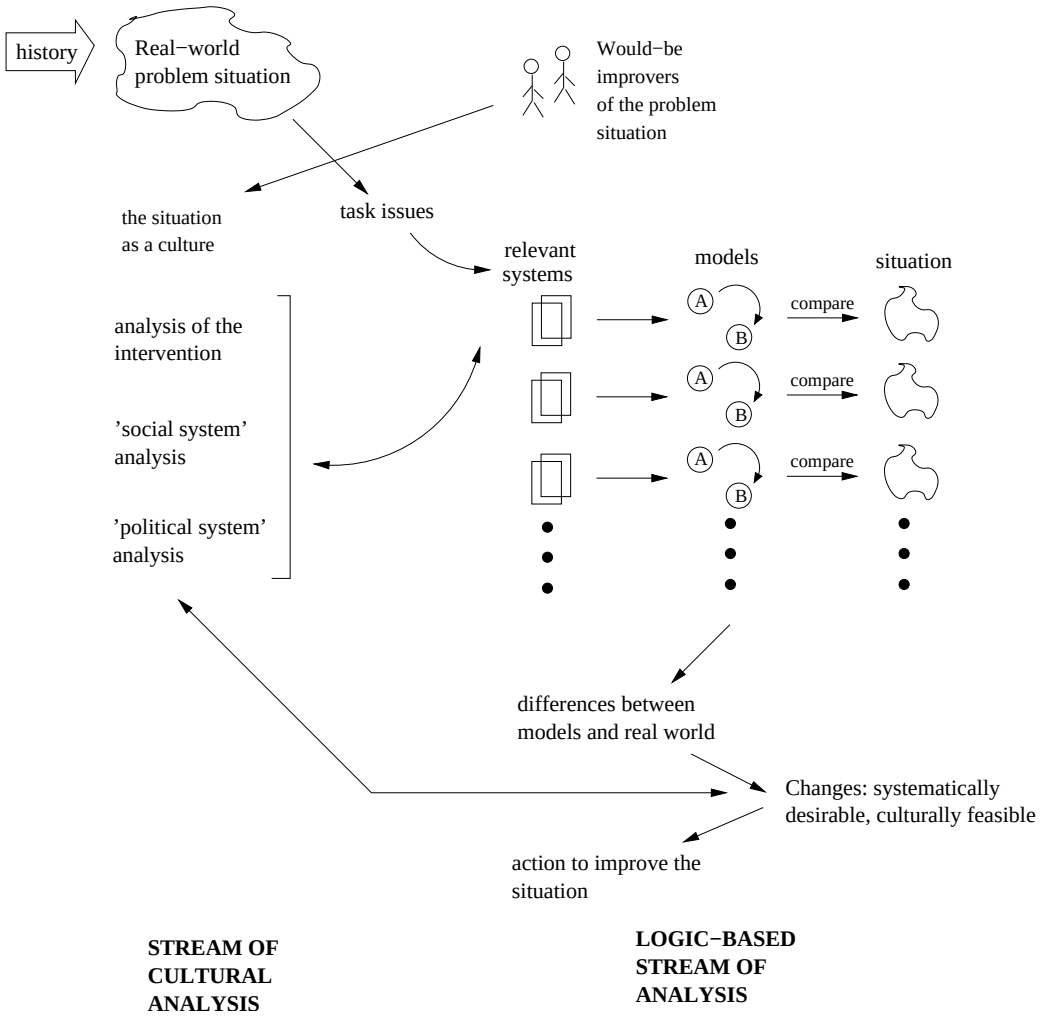


Figure 2.5: *The Process of SSM [Checkland and Scholes, 1990].*

Analysis One: Roles

In Analysis One, the involved roles in the problematic situation are considered. Checkland and Scholes describe three roles: Clients (persons who caused the SSM study to take place), problem owners (who could benefit from the desired change) and problem solvers (interventionists or users of SSM) [Checkland and Scholes, 1990]. Producing a list describing these roles of the problematic situation helps to select and model appropriate systems directed at improving the situation.

Analysis Two: The Social System

In order to design relevant systems, it is a good idea to take advantage of the existing knowledge on how the social context is functioning in the situation. Checkland and Scholes have proposed a model that is to be used in the analysis of the social system [Checkland and Scholes, 1990]. The model assumes a continually changing interaction between three elements: Roles, norms and values. *Roles* is related to social positions in an organization or community (system developer, project manager or senior manager). A person assuming a particular role is expected to behave in a way that reflects his/her particular social position. The expected behaviour resulting from a particular role is known as *norms*. *Values* is the last element of the model and relates to whether a person in a role is actually behaving according to local standards, i.e. *values*. The model is depicted in Figure 2.6.

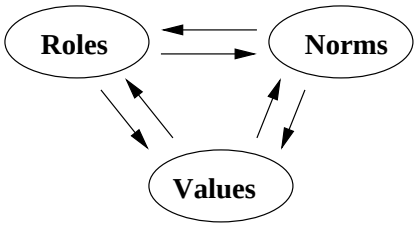


Figure 2.6: *The relationship between roles, norms and values used in Analysis Two [Checkland and Scholes, 1990]*

In SSM, the model is used to map the relationship between the three parts which leads to a notion on how the social system is currently organized.

Analysis Three: The Political System

The last part of the cultural analysis concerns an analysis of the political context surrounding the problematic situation. A political analysis of the situation can enrich cultural knowledge built up in Analysis One and Analysis Two. In effect, the understanding of the cultural analysis complements the activities within the logic-based analysis [Checkland and Scholes, 1990]. Checkland and Scholes propose that a political analysis is conducted through a series of power-oriented questions where power relates to examples such as formal authority, intellectual authority, personal charisma or external reputation. Based on the work of Stowell, Checkland and Scholes propose the following power-oriented questions that is to be answered [Checkland and Scholes, 1990, pp. 51]:

- What are the commodities through which power is expressed?

- How are these commodities obtained, used, protected, preserved, passed on, relinquished?
- Through what mechanisms?

With respect to these questions, it should be noted that Checkland and Scholes assume that power can be viewed as a commodity.

The cultural aspect of SSM is supposed to complement the logic-based thinking and is not a one-time analysis. The intention is to involve the understanding and knowledge of the cultural analysis continually, supplementing the logic-based thinking as depicted in Figure 2.5.

2.3.5 Transformation System vs. Interaction System

With respect to transformation systems, they only represent the transformation from an initial state (input) to a final state (output) Intermediate states in the transformation process are not considered. Mathiassen and Nielsen argue that the notion of transformation systems fitting all human activity systems is too narrow-minded. They instead propose *interaction* system as an alternative. Compared to a traditional transformation system, an interaction system is often relevant when processes (or actors) in a human activity system interact with a material [Mathiassen and Nielsen, 1998]. The following quote is from Mathiassen and Nielsen and expresses their definition and perception of both transformation systems and interaction systems [Mathiassen and Nielsen, 1998, pp. 338]:

> *"...transformation systems denote soft systems that: define an overall change. In contrast, we let interaction systems denote soft systems that: operate in an interactive fashion on some material (e.g., processes, actors, objects, information) and which focuses on achieving invariance in the state space (e.g., satisfactory distribution of resources, provision of relevant information, effective coordination between individual actors)."*

In order to further explicate the difference between the two complementary soft system concepts, Mathiassen and Nielsen propose the distinction expressed in Table 2.1.

The use of interaction systems instead of transformation systems results in slight modifications to the CATWOE; it is changed into a CAIWOE. Accordingly, the semantics of the CAIWOE is the following:

- ***C****ustomers*: The victims or beneficiaries of **I**
- ***A****ctors*: Those who would do **I**

	Transformation system	**Interaction system**
Identify	A relationship between well-defined input and well-defined output expressed as a process.	An interactive operation on some structural artifact or material expressed as a process.
Perspectives	Change of material without focus on intermediate states.	Invariance of material with particular focus on different intermediate states.
Examples	Organizational change Systems development Design & Intervention Construction Production	Management Information Administration

Table 2.1: *A distinction between two complementary soft system concepts taken from [Mathiassen and Nielsen, 1998]*

- ***I**nteraction process*: The interaction and relevant material
- ***W**eltanschauung*: The world view which makes this **I** meaningful in context
- ***O**wners*: Those who could stop **I**
- ***E**nviromental constraints*: Elements outside the system which it takes as given

This slightly modified semantics is based on Mathiassen and Nielsen's discussion regarding the adoption of the CATWOE in interaction systems [Mathiassen and Nielsen, 1998].

2.4 Experimenting with Prototypes

The results from the ongoing SSM analysis are to be used in experimenting with system ideas using prototypes. In general, prototyping aims at exploring, evaluating or experimenting with ideas and basic design features using unfinished models of the final application. Such models may even be in the form of mock-ups, i.e. cheap models made from paper, photographs, hand-drawn pictures, screen dumps, match-boxes etc. Either way, the intent is to answer essential questions about ideas, concepts, properties or design by experimenting with uncertain factors. Many definitions of a prototype exist. According to Pfleger, a prototype is defined as follows [Pfleger, 2001, pp. 51]:

*a **prototype** is a partially developed product that enables customers and*

> *developers to examine some aspect of the proposed system and decide if it is suitable or appropriate for the finished product.*

This definition emphasizes the importance of customer involvement and views the prototype as a partially developed *product*. However, a prototype may also be used for other purposes without involving the customer, e.g. serving as a basis for technological experiments. Furthermore, the prototype does not necessarily involve a *product*, but rather an *application*.

According to Budde et al., prototyping is characterized by the following two properties [Budde et al., 1992, pp. 89]:

1. *Prototyping* is an *approach* based on an evolutionary *view* of software development and having an impact on the development process *as a whole*.

2. *Prototyping* involves producing *early* working versions ("prototypes") of the future application and experimenting with them.

The prototype never represents the complete application, but rather individual components or selected features. By only focusing on parts of an application, prototyping enables experimentation with different ideas, models or design features. There is, however, one requirement. The development of a prototype must be rapid; the prototype should be easy to develop and to redesign in a matter of days or weeks. The basic idea is quickly to be able to test ideas and assumptions in order to find out how they work in practice, if they were the wrong ideas in the first place or if they just need modification. Through experiments, one wishes to draw conclusions as to the qualities and properties of the future application.

However, the use of experiments does not render traditional systems development methods superfluous [Thorshøj, 1989]. Rather, experiments should be used when:

- The degree of uncertainty is high
- New patterns of behaviour must be developed related to the applicability of computers and IT
- New machinery / programs must be put to use
- Central work routines must be supported

Following 'The Principle of Limited Reduction' proposed by Mathiassen and Stage [Mathiassen and Stage, 1992], described in Section 1.2, experiments may then be combined with more analytical approaches in order to address the complexity introduced by experimentation.

2.4.1 Goals of Prototyping

However, before embarking on a prototyping adventure, one needs to be aware of the differing goals of prototyping. These goals set boundaries for the basic objective of the experiments. In general, three overall goals of prototyping exist [Floyd, 1984], [Budde et al., 1992] and [Thorshøj, 1989]:

- *Exploratory prototyping:* This form of prototyping is used when ideas and concepts built into the prototype must be explored. Often, the problem at hand is somewhat unclear. Different aspects of the future application may be the focus of the prototype and the goal is to explore possibilities and options based on a variety of prototypes. Exploratory prototyping may be of particular interest in the initial phase of a development project when developers may explore various options in cooperation with users from the client organization.

- *Experimental prototyping:* The focus of this form of prototyping is on technical implementation of development goals. It is directed towards design of selected parts of an application based on a particular system idea. Users may refine their perception of what the application is about and how it is going to be used, focusing on requirements for the type of computer application. From the developers' point of view, experimental prototyping is an opportunity to go into detail with a specific design proposal and test the feasibility and suitability of the future application. Thorshøj mentions one problematic aspect of experimental prototyping [Thorshøj, 1989]: The design proposal is often presented early on in the development process without prior analysis of the application domain and without exploration of alternative proposals. This increases the risk of an inappropriate design or a dangerously narrow focus on unnecessary details, rather than on the general concept [Thorshøj, 1989].

- *Evolutionary prototyping:* This form of prototyping distinguishes itself from the two others by being part of the development process. Evolutionary prototyping is a continuous process aimed at stepwise adaptation of the application to "rapidly changing organizational constraints" of the user organization [Budde et al., 1992, pp. 93]. Through this process, the future application is developed in close cooperation between users and developers. Budde et al. notice that the developers' role changes from being authors of the final product to becoming technical consultants cooperating with the users to evolve the prototype into the future application [Budde et al., 1992]. In this way, evolutionary prototyping is closely linked to both iterative development approaches and incremental development approaches.

It is important to notice that none of these goals of prototyping rule out the others; one prototyping experiment may include various degrees of all three overall goals [Thorshøj, 1989].

2.4.2 Types of Prototypes

According to Budde et al., prototyping may assist in three major activities of the software development process: Initiating the project, analyzing the business needs and designing and constructing the software application [Budde et al., 1992]. In each of these activities, certain kinds of prototypes are helpful. Budde et al. distinguish between the following four kinds [Budde et al., 1992, pp. 91-92]:

- *Presentation prototype*: This kind of prototype supports the initiating of the project. Presentation prototypes may be used to clarify certain aspects of the specification before contracts are made and be submitted as part of the proposal for the client. This may be done to ensure either that client and software manufacturer share an understanding of e.g. user requirements, or to demonstrate the feasibility of the future application. Presentation prototypes are often built very rapidly and may break the conventional rules of good software engineering practices in order to decrease development time. In particular, presentation prototypes may be used in exploratory prototyping to visualize ideas and concepts for further evaluation.

- *Prototype proper*: This form of prototyping may concern any part of the future application. A prototype proper could be made to visualize the user interface or to demonstrate certain functionality. It is a temporary operational system which is constructed in parallel with the information systems model. Often, prototype propers are used to explore different options or experiment with various aspects of the application. They provide a concrete basis for evaluation of the problem solution being sought.

- *Breadboards*: This kind of prototype is aimed at clarifying development issues related to system architecture or functionality. The focus is on technical aspects of the future application. Therefore, future users do not participate in the evaluation of breadboards. The details of the breadboard are derived from the information systems model or the software specification. Breadboards are often used as part of experimental prototyping where technical experiments may be required to evaluate feasibility of solution proposals.

- *Pilot system*: This form of prototyping covers prototypes that are developed into at pilot system. Such a pilot system is eventually developed into the core of the future application after having reached a certain degree of refinement. The focus is not on experimental testing or illustrative purposes, but rather on developing parts of the future application without launching a full-scaled systems development project. Often, the pilot system may be developed to survey the feasibility of the chosen system solution and to gain some initial experiences before launching the full-scaled development effort. The pilot system is thus a result of an evolutionary approach to prototyping.

Regardless of the type of prototype used, experimenting with prototypes prepares for the adoption of the future application to the client organization [Budde et al., 1992]. During development, the users get a sort of preview of the future application, allowing them to prepare for the change involved in the transition from the old to the new situation. For the developers, each type of prototype is viewed as an executable specification which describes specific aspects of the future application. Such specifications enable evaluation of different models and furthermore reduce ambiguity during software construction [Budde et al., 1992, pp. 92].

In addition to the above types of prototypes, Pressman (among others, e.g. [Pfleger, 2001] and [Thorshøj, 1989]) addresses the concept of a 'throw-away' prototype [Pressman, 2000]. The question is whether of not to *keep* the prototype as part of the final application. In the 'throw-away' approach, only ideas and concepts are reused in the implementation of the future application. The prototype is used for specifying requirements and less focus is put on the technical implementation. Developers, who participated in the construction of the prototype, may quickly be able to re-construct the parts covered by the prototype. Such re-constructed design and implementation tend to be more clean and refined because of the experiences collected during the construction of the prototype. In the 'keep-it' approach, the prototype is incorporated into the future application [Pressman, 2000].

2.4.3 The Process of Prototyping

In order to guide the experiments using prototypes, some sort of process must be followed. The experiments must be conducted with a defined main objective which guides the prototyping process. This main objective acts as the motivation for the experiments and influences the specific stages involved in the process.

Various models of the prototyping process exist, but they all involve a set of basic stages such as planning, construction, experimenting and evaluating. We adopt the process described by Mathiassen et al. [Mathiassen et al., 1998]. Their description of the process of prototyping is based on five main stages which roughly translates to the following:

1. Planning
2. Development
3. Preparation
4. Experimentation
5. Evaluation

Each of these five stages will be addressed in the following paragraphs based on [Mathiassen et al., 1998] and [Thorshøj, 1989].

Planning

In the *planning* stage, the objective of the experiment must be precisely defined and delimited. Why are we conducting the experiments? and What is it that we want to find out? Furthermore, the content of the prototype must be determined. This may include a detailed set of requirements and a preliminary design of the prototype. Mathiassen et al. suggest answering the following three questions in order to clarify and motivate the content of the prototype [Mathiassen et al., 1998, pp. 43]:

- What is the focus of the experiment?
- What falls outside of our interests?
- Which conditions must be satisfied?

In addition, initial plans are made for the subsequent stages, thereby developing schedules and setting deadlines. It is stressed by Thorshøj that the experiments must be carefully organized and systematically performed [Thorshøj, 1989].

Development

The *development* stage constitutes the stage where the actual prototype is constructed. The level of detail and accuracy during the requirements analysis and the design of the prototype may vary. In some cases, both analysis and design may be done only very abstractly in favour of a very rapid construction of the prototype. However, the prototype must be constructed based on the main objective of the experiment [Thorshøj, 1989]. In this respect, it is the main objective along with its sub-objectives that guides the development and determines the content of the prototype. Furthermore, it is the experimentation, not the prototype, that is the purpose of prototyping.

Another point to be made regarding the construction of the prototype concerns the content. Budde et al. (among others, e.g. [Thorshøj, 1989] and [Pressman, 2000]) distinguish between two approaches to the construction of prototypes. If software construction is viewed as the design and implementation of a number of layers, e.g. model, functionality and graphical user interface as depicted in Figure 2.7, then it is possible to distinguish between horizontal prototyping and vertical prototyping [Budde et al., 1992].

Horizontal prototyping addresses the implementation of one layer, e.g. the graphical user interface. Required functionality from the other layers are ignored and omitted. In the case of graphical user interface prototyping, no functionality is implemented, but rather, a set of screens are created along with forms and menus to give the user an impression of what the future application is going to be like. In contrast, vertical prototyping focuses on the implementation of selected parts of the future

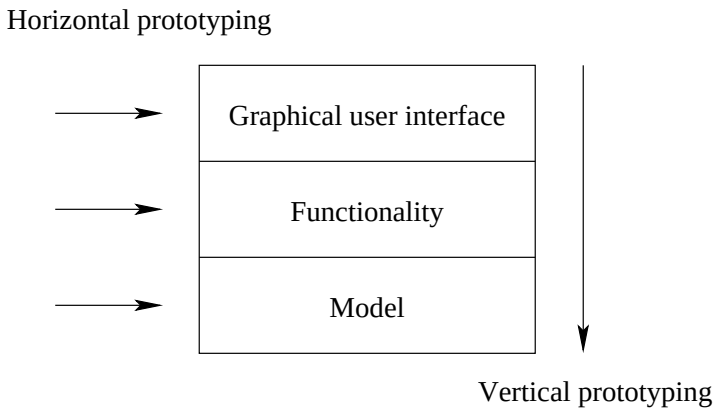


Figure 2.7: *An example of a layered component structure.*

application. Such implementation covers the required functionality from all layers in order to implement the selected part.

In order to satisfy the requirement of rapidness in the development of prototypes, 4th generation programming tools may be used. We refer to [Pressman, 2000] for an elaboration on various categories.

Another aspect of the development concerns the documentation of the prototype. Such documentation could for instance include descriptions of system architecture and functionality along with justifications for the choice of programming languages or tools used in the development of the prototype.

Preparation

In the *preparation* stage, the actual experiments with the prototype must be prepared. This may include the writing of use cases or user stories to be utilized during the experimentation. Furthermore, the experiments may include future users so the degree of realism must be decided on. Mathiassen et al. suggest three conditions which need to be addressed: Cooperation, realism and selection [Mathiassen et al., 1998]. The *cooperation* between users and systems developers must be defined in order to establish their roles during the experiment. For instance: Is the testing based on observation where systems developers observe while the users test the prototype? or May the systems developers interact more freely with the users? Trigg et al. discuss the notion of open-endedness in relation to cooperative prototyping [Trigg et al., 1991] (see Subsection 2.4.5 for an elaboration of cooperative prototyping). They advertise for a more loosely conducted experimentation stage without restrictive boundaries. Relying on an open-ended interaction style between users and developers stimulates the creative process, creating a more relaxed setting for the generation of ideas. Event hough their results only relate to

cooperative prototyping, their point may still be useful in other forms of prototyping.

Realism may concern both the environment and the situation in which the prototype is tested. The experiment could be set up in a fictitious environment away from the actual environment where it is going to be used. Bondgård et al. use the terminology *integrated* or *detached* to characterize the setting of the experiment [Bondgård et al., 1990a]. Either the prototype may be experimented with as an integrated part of the user's work environment or the experiment may take place detached from the context in which the future application is going to be used. In both cases, several issues must be dealt with. If the experiment is integrated into the user's work situation, there may be various events taking place simultaneously, making it difficult to document. On the other hand, if the experiment is detached, e.g. taking place in a usability laboratory, the developers have more control over the situation and may introduce suitable stimulus in order to observe the user's reaction. Such direct correlation between stimulus and reaction may not be evident if the experiment is conducted as an integrated part of the users' work environment. However, such control is gained at the expense of realism.

Finally, it is necessary to *select* the users who are going to participate in the experiments. Their level of preparation must also be decided on; if the users need to prepare for the experiment, they must know beforehand. Thorshøj states that the participants must understand and accept both the use and role of prototyping [Thorshøj, 1989]. He mentions that users may be quite unfamiliar with commenting the use of software applications. Furthermore, the users may have difficulties accepting the fact that the prototype contains errors and shortfalls. They might not be able to abstract from this fact, leading to a problematic evaluation. With respect to the number of participating users, Saarinen and Sääksjärvi argue that the quality, not the quantity, of user participation is of key importance [Saarinen and Sääksjärvi, 1990] to the success of an information systems development project. In Subsection 2.4.5, we elaborate on this argument. Therefore, careful considerations must be put into the selection of participants.

Experimentation

In the *experimentation* stage, the prototype is experimented with in the prepared environment. During experimentation, the results must be documented for further investigation, e.g. by taking notes, writing diaries, audio or video recording the experiment. With respect to running the experiments, Thorshøj describes six overall approaches [Thorshøj, 1989, pp. 12-13]:

- Demonstration
- Cooperative evaluation
- Cooperative construction

- User evaluation
- Operating evaluation
- Parallel operating

The first three approaches cover various degrees of user involvement in the prototyping process whereas the last three approaches relate to the realism of the experimentation. We refer to the subsection on user involvement in prototyping, Subsection 2.4.5, for a description of the first three degrees of user involvement.

Evaluation

In the *evaluation* stage, the results are analyzed and compared to the objective set out during planning. The result could be a new set of areas that needs to be explored, e.g. new ideas or changes to the prototype, or a draft for a preliminary design. Either way, the evaluation stage is used to plan what to be done next. Thorshøj stresses that an experiment is a mutual learning process. The users learn about how the future application is going to affect their work. During experiments, users are confronted with the prototype and questions are asked by the developers. Hereby, the users are forced to reflect on their own work routines, possibly involving the entire organization [Thorshøj, 1989]. The prototype thus discloses a perception of a future use situation that may be of vital interest in the development of the future application. During prototyping, the developers learn about the users' wants and needs along with organizational issues. The result of this two-way learning process is new knowledge. This mutual learning process minimizes the gap between users' and developers' perception of the requirements, ensuring a more tailored future application.

2.4.4 Iterativeness of the Prototyping Model

It is important to notice that prototyping is often not a linear process as suggested by the description of the five stages in the previous subsection. Rather, it is an iterative process where a new experiment is conducted based on the results from the previous experiment. The results provided in the evaluation stage may spin off new ideas, which need to be tested, or disclose areas of concern that the systems developers have not thought about. Either way, the process continues until a satisfactory result is reached and a conclusion can be made from the conducted experiments. However, the iteration does not necessarily involve all of the five stages. For instance, an evaluation may suggest that further experimentation is needed in order to address the objectives set out in the planning stage. In this case, the iteration may start off by preparing the experiments.

The iterativeness of the prototyping process is illustrated in Figure 2.8.

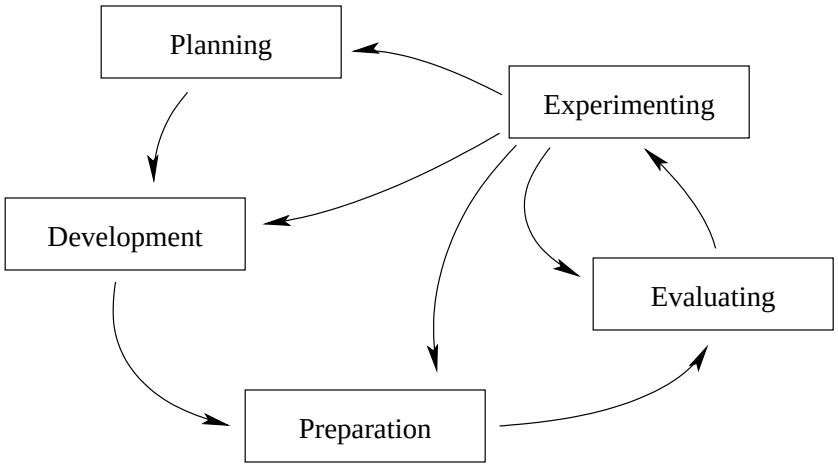


Figure 2.8: *Iterativeness of the prototyping process.*

2.4.5 User Involvement in Prototyping

The involvement of users in the prototyping process is essential to the development of a satisfactory product. The overall goal is to ensure quality in the future application. However, various degrees of user involvement may be adopted. Thorshøj describes the following three approaches [Thorshøj, 1989]:

- Demonstration
- Cooperative evaluation
- Cooperative construction

These approaches are elaborated on in the following. In addition, we afterwards take a brief look at Kaj Grønbæk's notion of *cooperative prototyping* along with Ehn's concept of a *shared language game* involving both users and developers.

In a *demonstration* of a prototype, the users are not actively involved in the experiments, but rather observe as the features of the prototype are demonstrated to them. Thorshøj points out that demonstration puts less stress on the users, making them more comfortable before they themselves get the chance to try out the prototype.

In *cooperative evaluation*, users and developers evaluate the prototype in cooperation during the experimentation. The users may comment the prototype, e.g. by 'thinking out loud', while developers guide the experimentation. The developer's role is either observant or actively participating, asking questions along the way and discussing the prototype with the user. The topics for the cooperative evaluation may either be strictly prepared, e.g. having the user go through a set of use cases, or more loosely prepared, taking on a more open-ended form as advertised by Trigg

et al. [Trigg et al., 1991] in relation to Grønbæk's cooperative style of prototyping presented separately in the following paragraph.

In *cooperative construction*, the users are involved in the design and construction of the prototype. This may set requirements as to the development tools used during the experiment session. Such tools must support 'on-the-fly' manipulation of the prototype. Cooperative construction is advertised by Grønbæk's notion of 'cooperative prototyping' addressed separately in the following paragraph.

Cooperative Prototyping

One rather radical form of user involvement in prototyping is described by Kaj Grønbæk [Grønbæk, 1989]. He introduces the term *cooperative prototyping* covering the active involvement of users in the prototype design. Here, development and experimentation are closely linked because they are performed cooperatively, involving both users and developers. Based on an initial proposal, the prototype is refined in close cooperation between users and developers as the users participate creatively in the design process, making suggestions for improvements. Cooperative prototyping is based on the assumption that "active user involvement in design is the key to anticipate the problems traditionally seen when a new computer application is installed in a work-place [Grønbæk, 1989, pp. 220]. The key to the approach is rapid modifications. The users must be able to see the results within minutes in order to assess and evaluate the changes made. In this respect, computer-based tools to support such rapidness are of vital importance to the success of cooperative prototyping. Grønbæk emphasizes that such tools should support direct manipulation, making way for immediate modifications [Grønbæk, 1989].

The basic idea of cooperative prototyping is to set up sessions based on work-related tasks from the use domain. Various models, mock-ups or prototypes may be used in these sessions. Based on experimentation with work-related tasks, improvements may be suggested or even tried out on-the-fly, creating a highly dynamic design process. In this way, development, experimentation and evaluation are combined to form a coherent design satisfying the user needs. In relation to cooperative prototyping, Trigg et al. advertise for a more open-ended approach [Trigg et al., 1991]. By not imposing too much structure on the experiment sessions, the users are provided with a high degree of freedom to take charge of the prototype. Their responses arise freely and may disclose unforeseen aspects of the prototype [Trigg et al., 1991].

'Shared Language Game'

Another aspect of user involvement concerns the communication between users and developers. Saarinen and Sääksjärvi emphasize the importance of good communication skills in their study on user involvement in information systems development projects [Saarinen and Sääksjärvi, 1990]. In particular, good communication skills

on the part of the developer is one of the most important attributes for a successful outcome of the project [Saarinen and Sääksjärvi, 1990, pp. 35]. Such reasoning may show equally applicable in prototyping.

It is thus important for both users and developers to be able to understand each other in order to fully benefit from the collaboration and not misinterpret statements made by the counterpart. In addition, as prototyping may be viewed as a process of mutual learning [Thorshøj, 1989] and [Trigg et al., 1991], the work of Ehn [Ehn, 1989] is important in understanding the communication between participants in the experiments [Bondgård et al., 1990b]. Ehn introduces the concept of a *shared language game* [Ehn, 1989] which he uses to analyze the communication and knowledge transfer between users and developers.

He argues that neither the users nor the developers may be able to explicate opinions and viewpoints to the other. In order for this to happen, they need to understand the background of their counterpart and their *language*. In this respect, language should not be taken literally, but rather be interpreted as a combination of spoken language, body language, sounds and noises, made along with the context of the situation.

On one hand, the developers need access to the practical understanding of the users' work in order to provide support via the future application. However, such practical understanding of skills involved in the users' work may be hard to explicate. They are what Polanyi terms *tacit knowledge* [Polanyi, 1962]. In other words: "What the users can tell about their work differs considerably from the knowledge involved in doing it" [Bondgård et al., 1990b, pp. 15]. On the other hand, users must be able to understand the developers and their intentions; they must participate in the language game of the developers in order to understand technical contributions and ideas proposed by the developers. The prototype may in this respect serve as a means of communication, supporting the shared language game.

Chapter 3

Research Setting

Having addressed our research approach of action case along with the two methodologies that we are going to combine in our efforts to tailor knowledge management systems, we now address the setting in which our current research takes place. In particular, we review important aspects of the first part of our research conducted during the previous research period and documented in [Hosbond and Ørtoft, 2003].

The first part of our research took place in the two Western sub-divisions of the Systems Development branch of WM-data Denmark during the previous research period. The focus was stated in the following research question:

"*How does WM-data pursue knowledge management activities in their current software practices?*"

In this chapter, we start off by shortly providing an overview of the theoretical knowledge management foundation of the current study. The section is meant as a summary of relevant theories from our previous study that we have also put to use in the present study. Therefore, it does not go into depth with the particularities of each theory, but rather it introduces terminology used throughout this study. Next, we describe the setting in which our research from the previous period took place. After that, we summarize key characteristics of WM-data, drawing upon the results of our analysis and discussion from [Hosbond and Ørtoft, 2003]. The intent of revising these characteristics is to motivate the focus of the current research period: P^+. It will be argued why the P^+ initiative is of importance seen from a knowledge management perspective. Relevant theories within knowledge management summarized in Section 3.1, along with our knowledge gained during the previous study serve as a basis for such a discussion. Furthermore, P^+ will be described along with changes made in the organizational setting that are important to our research. First, we turn to the theories within knowledge management.

3.1 Theories on Knowledge Management

In today's society the creation and exploitation of knowledge has received great attention within the past decade. The organizational focus on intellectual resources calls for at more efficient management of knowledge in order for organizations world-wide to be able to keep up in todays highly competitive business markets. Broadly defined, knowledge management addresses the use, sharing, acquisition, capturing and creation of knowledge [Kautz and Thaysen, 2001]. In this section, we present some of the relevant knowledge management theories that we applied in our previous study [Hosbond and Ørtoft, 2003] and the present study. Moreover, the following elaboration will be based on our understanding of the theoretical material in question. For a more detailed elaboration on the knowledge management theory presented in this section, we refer to [Hosbond and Ørtoft, 2003, Chapter 3] and the original articles and books.

3.1.1 Definition of Knowledge

In our previous study, [Hosbond and Ørtoft, 2003], we discussed several definitions of knowledge including the distinction between data, information and knowledge. Nonaka, defines knowledge as a *personal* justified true belief. [Nonaka, 1994].This perception of knowledge contradicts the traditional epistemology of knowledge where knowledge is considered to be a *justified true belief* [Nonaka, 1994]. The difference lies in whether knowledge is considered universally true and expressible by means of formal logic (hard systems thinking) or considered personal to the individual and a result of personal beliefs, skills and experiences (soft systems thinking). We believe knowledge to be personal and therefore adopt Nonaka's definition.

On an overall level, Nonaka distinguishes between two types of knowledge: Tacit and explicit [Nonaka, 1994], described in [Hosbond and Ørtoft, 2003, Section 2.3]. These two types of knowledge form the basis of his theory on organizational knowledge creation briefly addressed in Subsection 3.1.2. In relation to software organizations, Mathiassen et al. propose another distinction by combining Nonaka's tacit-explicit dimension with Hansen et al.'s informal-codified dimension (Mathiassen et al. rename 'personal' to 'informal'): Situated, exemplary, community and procedural knowledge [Mathiassen et al., 2003]. *Situated* knowledge refers to the unique knowledge that is created due spontaneous situations in which people informally interact and communicate. It is both tacit and informal. *Exemplary* knowledge is explicit and informal knowledge found in project documentation. The outcome of a meeting may be documented which then represents exemplary knowledge. Contrary, to exemplary knowledge, *community* knowledge is tacit and represents the tacit, but codified rules of a group, e.g. group of systems developers. Community knowledge represents the unique characteristics of a group. Finally, *procedural* knowledge is explicit and codified and is represented by formalized notations, techniques and disciplines.

3.1.2 Organizational Knowledge Creation

Nonaka's theory on organizational knowledge creation centers on the four knowledge conversion modes of the SECI model: *S*ocialization, *E*xternalization, *C*ombination and *I*nternalization [Nonaka, 1994]. The basic idea is that knowledge is created through continuous conversion between tacit and explicit knowledge. *Socialization* is the process of creating tacit knowledge through human interaction by means of a shared context. The shared context is established through shared experiences. In the process of *Externalization*, tacit knowledge is converted into explicit knowledge by adopting metaphors to express tacit knowledge. Metaphors have the capability of visualizing tacit knowledge through stories, e.g. stories or incidents from daily practices. *Combination* address the systematization of explicit knowledge. That is, combination of existing explicit knowledge may reveal new explicit knowledge. The process of *Internalization* represents the conversion of explicit knowledge to tacit knowledge. The best way to turn explicit knowledge into tacit knowledge is through action. Learning through *hands-on* experience is therefore important in order to internalize knowledge.

The four conversion modes and their interrelations are illustrated in Figure 3.1.

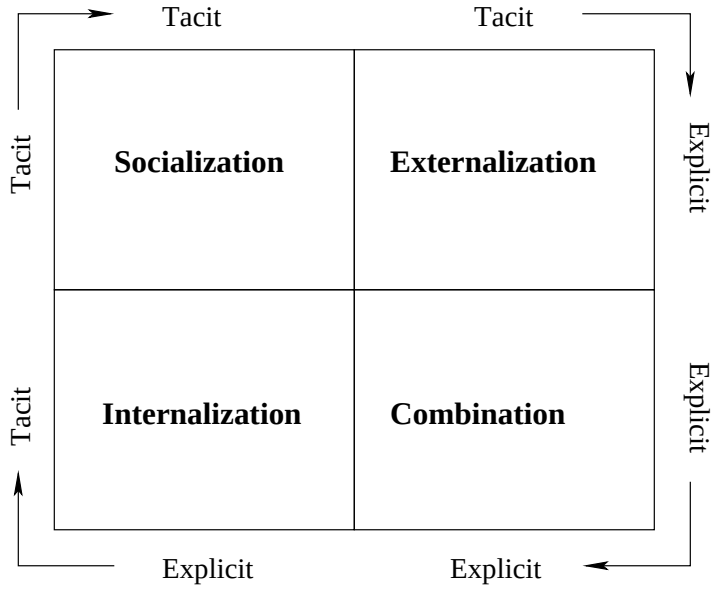


Figure 3.1: *The SECI model proposed by Nonaka [Nonaka, 1994].*

The Concept of *Ba*

In addition to organizational knowledge creation, Nonaka has also been involved in work together with Konno addressing the context in which knowledge is created

[Nonaka and Konno, 1998]. This context is termed *ba* and may be thought of as a "*shared space for emerging relationships*" [Nonaka and Konno, 1998, pp. 40]. Briefly characterized, the shared space may have one or more of the following characteristics [Nonaka and Konno, 1998]:

- Physical (such as offices, class rooms or lunch halls)
- Virtual (such as e-mails, teleconferences, Internet chat rooms or newsgroups)
- Mental (such as shared experiences, ideas or ideals)

Ba is the context in which knowledge is created and shared and is closely tied to the four conversion modes of the SECI model. Therefore, Nonaka and Konno present four different *ba*s, one for each conversion mode [Nonaka and Konno, 1998]. The interrelations between the four *ba*s and the conversion modes are illustrated in Figure 3.2.

Originating Ba (Socialization)	**Interacting Ba** (Externalization)
Exercising Ba (Internalization)	**Cyber Ba** (Combination)

Figure 3.2: *The different bas and their relationship to the knowledge conversion modes of the SECI model.*

Originating ba is the context for the socialization phase and is the shared space for individual face-to-face transfer of tacit knowledge using all of the physical senses. Experiences, mental models and feelings are shared in order to facilitate knowledge conversion. The *Interacting ba* relates to the conversion mode of externalization and is therefore a shared space in which tacit knowledge can be converted to explicit knowledge. The conversion is primarily fostered by dialogue and metaphors are used to explicate mental models. *Cyber ba* represents the context in which the combination phase takes place. *Cyber ba* relies on information technologies such as group-ware (e.g. Lotus Notes) and written documentation. The aim is to systematize and combine explicit knowledge. *Exercising ba* relates to the internalization phase and is intended to build a context in which initiatives based on *hands-on* experience and learning-by doing are conducted.

3.1.3 Knowledge in Communities-of-practices

In our previous study, it was argued that the project culture at WM-data closely resembles that of communities-of-practices. Therefore, an in-depth understanding of such communities was sought through the work of Brown and Duguid [Brown and Duguid, 1991] and Mathiassen [Mathiassen, 1998].

In particular, Brown and Duguid take on a practical approach in describing organizational learning, viewing communities-of-practices as the starting point. In the words of Mathiassen, communities-of-practice are created from practitioners collaborating and sharing work experiences [Mathiassen, 1998]. Brown and Duguid propose a unified view of organizational learning constituting three aspects: *Working*, *learning* and *innovation*.

Working comprises two types of practices: Canonical and non-canonical. A canonical practice is the expected practice of a community, i.e. the practices that the community is believed to have. The expected practice is often described in abstract terms and formed by upper management. In contrast, the non-canonical practice refers to the actual practices that a community follows. Brown and Duguid suggest three interrelated categories of *working*: Narration, collaboration and social construction relating to different aspects of work practices. Narration is extensively used by practitioners for sharing experiences. Such experiences are shared by means of stories, emotionally referred to as as *war stories*. Such stories foster the creation of situated knowledge [Mathiassen et al., 2003]. Furthermore, when practitioners share experiences they collaborate and thereby foster collective learning. Being able to share experiences and create unified practices requires that a shared context is established. This is referred to as social construction and is related to the concept of *ba* proposed by Nonaka and Konno [Nonaka and Konno, 1998].

Learning is related to *working* and emphasizes the context and argues that knowledge must not be separated from practice, implying that learning is viewed as a social construction taking place through *working*. To ensure learning, trainees (less knowledgeable) must therefore have access to practitioners (someone knowledgeable) in order to observe their actions.

Innovation is based on the idea that non-canonical practices are essential to the innovation process. Non-canonical communities are less rigid and more dynamic in nature and therefore creates an inventive environment.

The three aspects of working, learning and innovation serves as a useful perspective on organizational learning and communities-of-practices that is very useful in providing an understanding of systems development practices. For further elaboration of the three aspects of working, learning and innovation, we refer to [Hosbond and Ørtoft, 2003] and Brown and Duguid [Brown and Duguid, 1991].

3.1.4 Knowledge Management Strategies

Another aspect of knowledge management is the strategic aspect. Hansen et al. argue for two overall strategies of knowledge management: Codification and personalization [Hansen et al., 1999]. In the codification strategy, companies manage their knowledge through the use of information systems and computers. Knowledge is made explicit, codified and stored in databases and documents, making it accessible to everyone in the organization. In this way, relevant knowledge may then be searched, retrieved and reused by others. One example of the codification strategy is the *Experience Factory* presented by Basili et al. where knowledge is packaged and stored as commodities [Basili et al., 1994]. Hansen et al. argue that the codification strategy is mostly employed by companies developing standardized products. The codification of knowledge enables such companies to efficiently reuse codified knowledge [Hansen et al., 1999].

In contrast, the personalization strategy focuses on person-to-person interaction in order to transfer knowledge. Information systems and computers are only used to facilitate communication between individuals. Terms like *networking* [Swan et al., 1999] and spontaneous communication are important to the personalization strategy as they provide a means of sharing knowledge. As opposed to the codification strategy, it is argued by Hansen et al. that the personalization strategy is mainly adopted by companies which provide tailored products. The tailoring of products limits the degree to which knowledge can be reused in the development of products.

In choosing between the codification strategy and the personalization strategy, it is thus important that the characteristics of the company is taken into consideration. Hansen et al. propose answering the three following questions in order to make the overall decision [Hansen et al., 1999]:

1. Do you offer standardized or customized products?

2. Do you have a mature or innovative product?

3. Do your people rely on explicit or tacit knowledge to solve problems?

In addition, Hansen et al. argue that companies must balance their strategy, not just following one of the two strategies. However, the two strategies should not be balanced equally, either. Rather, a 80-20 split should be employed, meaning that a company should strive for one main knowledge strategy used 80% of the time and a secondary knowledge strategy used 20% of the time, supporting the main strategy [Hansen et al., 1999].

3.1.5 Developing a Knowledge Strategy

Another strategic aspect of knowledge management is determining how to adjust the knowledge strategy to the business strategy of the company. This means that the knowledge strategy should reflect the overall business strategy of the company. Zack argues that forming a knowledge strategy requires insight into two aspects [Zack, 1999]: What type of knowledge does the organization currently possess? and What is the impact of current knowledge on the organization's ability to compete? With respect to the first aspect, Zack proposes three different knowledge levels: Core, advanced and innovative. These terms are almost self-explanatory, but for further elaboration, we refer to [Hosbond and Ørtoft, 2003, Subsection 2.5.3] or the original article by Zack [Zack, 1999]. In addition to these knowledge levels, we proposed a partitioning of WM-data's knowledge into three overall categories in our previous study [Hosbond and Ørtoft, 2003]: Technical, process-related and customer-related. These categories were used to determine the types and levels of knowledge possessed by WM-data.

With respect to the competitive strategy, Zack proposes making a map of the company's knowledge levels opposed to the knowledge levels of competitors. Such mapping should be based on the framework shown in Figure 3.3.

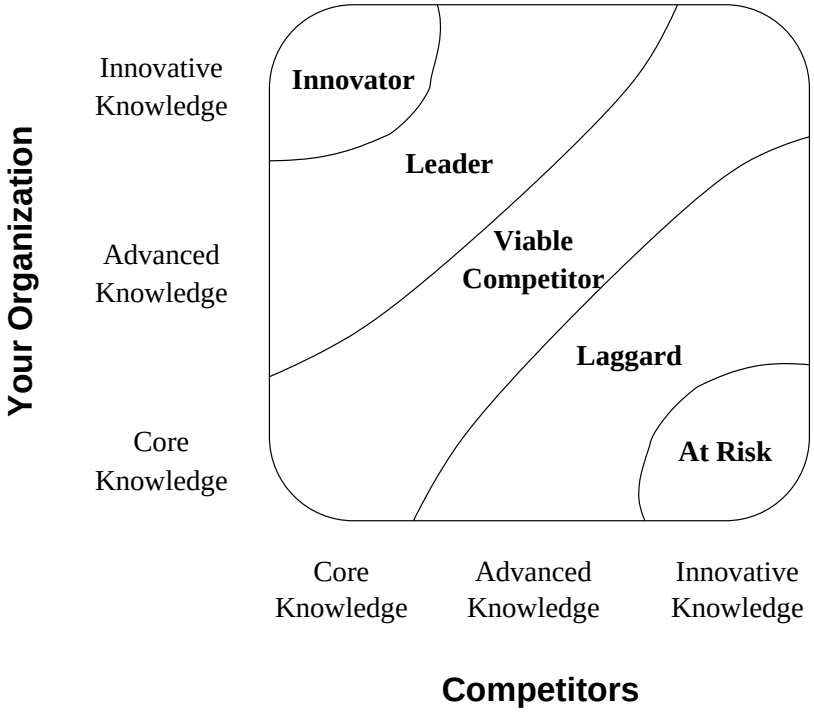


Figure 3.3: *The knowledge map proposed by Zack [Zack, 1999].*

In this way, companies may determine aspects of their business to support via knowledge management and accordingly determine the area of focus.

3.2 WM-data

WM-data is originally a Swedish company founded in 1969 by Thord **W**ilkne and Hans **M**ellström, hence the name **WM**-data. In the beginning of the seventies (1970-1976) they expanded into different regions of Sweden. During the last part of the seventies, the company turned international by expanding to other Nordic countries, including Denmark. By 1985, WM-data was quoted on the stock exchange, employing approximately 275 people. Through the eighties and the nineties, WM-data grew considerably. At the latest count (2001) they employed around 7,000 people in Sweden, Finland, Norway, and Denmark, having a turnover in 2001 of approximately SEK 12 billion.

3.2.1 The Organization

At the international level, the company's branches are run almost independently, only collaborating when customers do business across international borders. The focus of our work in the first research period concerned the Project Center West (PCW, for short) and Consultants Center West (CCW, for short), lead by John Kammersgaard and Dorte Gade, respectively. PCW and CCW are located as sub-divisions of the Systems Development branch as seen from the section of the organizational hierarchy shown in Figure 3.4.

Around 270 employees are working in the Systems Development branch at WM-data Denmark managed by Torben Iversen. The two Western sub-divisions, PCW and CCW, span two geographical locations in Aalborg and Aarhus, employing 40 and 97 people, respectively.

3.2.2 Key Characteristics

The history of PCW and CCW dates back to a Danish software company called Mentor Informatik. WM-data overtook Mentor Informatik in 1987, incorporating both the Aalborg department and the Aarhus department into its organizational structure. The backbone of Mentor Informatik consisted of a group of former candidates from the University of Aarhus. This influenced the company culture of Mentor Informatik that was very much informal because these people knew each other from their time at the university. This resulted in a very committed group of people employing a very informal communication style. Much of this original organizational culture has persisted since the takeover. Even though the organization may seem very hierarchical as depicted in Figure 3.4, the people we spoke with considered it to be very flat,

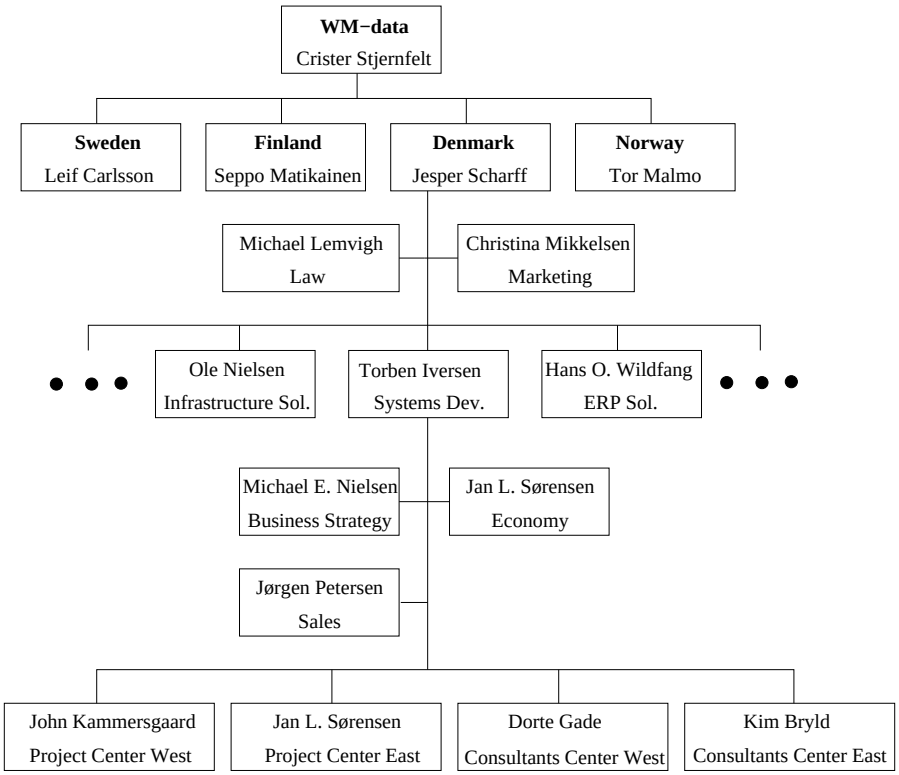


Figure 3.4: *A section of WM-data's organizational hierarchy showing the Systems Development branch of WM-data Denmark as of 2002.*

having immediate access to middle-management [Hosbond and Ørtoft, 2003]. This included a very informal relationship to both project managers and group leaders. At WM-data the project managers do not manage human resources. The welfare of employees are tended to by group leaders.

The analysis from our previous research period summarized a set of key characteristics in a company profile, see Table 3.1, containing seven characteristics of WM-data [Hosbond and Ørtoft, 2003]. Each of the characteristics in Table 3.1 will briefly be described in the following.

Much of the work-related communication in WM-data takes place as individual face-to-face communication. Project members often rely on direct and immediate access to co-workers, including the project manager. Such a way of communicating establishes *personal networks* around the organization as people work together in different projects. Most members of the organization rely on such personal networks when working out problems or asking for advice. However, *projects* also have a tendency to be very *isolated* from each other in the respect that knowledge created in one project is hardly ever reused by other projects. In most cases, such knowledge remains as

- Personal networks
- Isolated projects
- Grassroot SPI efforts
- Team spirit
- Informal communication
- Customized products
- Customer relations

Table 3.1: *Company Profile of WM-data [Hosbond and Ørtoft, 2003].*

individual tacit knowledge that is private to each of the project members.

Another characteristic of WM-data is the *team spirit* which prevails in projects. Each project member shows a great deal of commitment to the development task at hand. The most obvious example concerns overtime. Several of the interviewed project managers, as reported in [Hosbond and Ørtoft, 2003], noticed that it is not necessary for them to demand overtime as project deadlines approaches; project members willingly put in the hours necessary to complete the job.

The term *grassroot SPI efforts* stems from the fact that although WM-data has several company-wide initiatives, e.g. "Ratten" and ACE, the adoption of such initiatives is left to key individuals around the organization. Furthermore, SPI initiatives are not run top-down, but rather driven by the same key individuals. Our analysis from the previous research period showed that such efforts often lacked organizational anchoring and commitment from upper management. The result was that several initiatives were not fully taken advantage of because of lack of commitment.

The *informal communication* predominant in WM-data is closely tied to the informal company culture. When employees have immediate access to people around them, the communication becomes more informal. Formalities are omitted because they are viewed as rigid and unnecessary. When people need help they just ask.

The two last characteristics concern the business environment surrounding WM-data. The Systems Development branch of WM-data base their revenues on developing customized products to their customers. These are primarily the Danish government, to which they develop large administrative systems, and large Danish companies such as Egmont and Berlingske Bladhus [Hosbond and Ørtoft, 2003]. Because WM-data's business primarily is based on developing customized products, the *customer relations* become very important in the attempt to establish a permanent relationship to customers. Therefore, WM-data emphasizes the importance of the relationship to their customers. It is pointed out that about 90% of sales comes from sales within existing projects [Hosbond and Ørtoft, 2003]. However, the customer-related aspect also concerns the knowledge created about the customer and his business. Our analysis and discussion from our previous research period point out that customer-related knowledge becomes WM-data's competitive edge.

Besides the company profile, our analysis and discussion lead to other interesting characteristics of WM-data with respect to knowledge management. First of all, most knowledge in WM-data is held as individual tacit knowledge. Some knowledge concerning the way projects are run is, however, shared among project members; it is somewhat collective. The knowledge built up about the customer in each project may also be said to be be collective in the sense that everyone on the project shares a basic knowledge about the customer and the relation between the customer and the systems being built. Only very little knowledge is codified. Examples include "Ratten", the "delivery model" and essential project documentation such as requirements definition and requirements specification. The reliance on (individual) tacit knowledge is mainly due to the informal company culture predominant at WM-data.

Second of all, our discussion showed that WM-data puts very little focus on process-related knowledge. Although some SPI efforts have been initiated, most efforts are grassroot-driven as indicated by the company profile. Third of all, a comparison between WM-data and the characteristics of communities-of-practices as described by Brown and Duguid [Brown and Duguid, 1991] and Mathiassen [Mathiassen, 1998], showed that the project groups at WM-data closely resemble such communities-of-practices. This analysis provided a more in-depth understanding of the software practices as they take place in WM-data.

3.3 Choosing Our Focus

Our work from the previous research period identified several areas of WM-data's software practices that are interesting from a knowledge management perspective. Some of these areas include ongoing activities such as experience groups, "Ratten", ACE, cookbooks, and P^+ [Hosbond and Ørtoft, 2003]. Other areas contain aspects of current practices that are considered problematic, including the sharing of knowledge between project members that are separated geographically and managing SPI processes. All of these areas were analyzed and discussed with respect to selected theories within knowledge management, SPI and software process knowledge, see [Hosbond and Ørtoft, 2003]. In particular, one initiative stood out: P^+. In the following, we will describe the P^+ initiative in its current form along with its organizational setting. In addition, we argue why we find P^+ interesting and touch upon problematic aspects of P^+.

3.3.1 The P^+ initiative

P^+ is a group of highly experienced project managers who have a long history in WM-data. P^+ was established specifically to address the sharing of experiences between projects and act as a support for inexperienced project managers. The basic idea is that a group of experienced project managers act as collaborative partners

to project managers around the organization. Experiences and concerns may thus be shared via the P^+ group. All in all, the P^+ group consists of 6 project managers representing all geographical locations within the Systems Development branch (one in Aalborg, two in Ballerup, and three in Aarhus). The P^+ members only use part of their time working within P^+. The rest of the time they run their own projects as regular project managers.

The two main goals of P^+ are thus the following: Facilitating the sharing of experiences between projects and acting as support for inexperienced project managers. These goals are addressed through four main activities: P^+ meetings, sparring sessions, startup seminars, and theme seminars. Each of these four activities will be described in the following.

P^+ Meetings

P^+ *meetings* are held approximately once every 2 or 3 months, taking place at a different location each time. During meetings, issues that have been mentioned during sparring sessions may be discussed if they are considered important enough to be brought up. In this way, problematic aspects are brought to the attention of the P^+ members who may then deal with them and determine if they concern other projects. Such aspects may be either process-related or related to any particularities of project management. If the problematic aspect is important and general enough, the knowledge gained from having dealt with it may then be passed on to other project managers either via individual sparring sessions or via initiating SPI efforts.

Sparring Sessions

Sparring sessions are the actual meetings between a P^+ member and a project manager. Each P^+ member is assigned a number of project managers to whom he acts as a sparring partner. The frequency of sparring varies according to demands and needs. The basic idea with the sparring activity is that project managers get the chance to bounce ideas off of the P^+ member and to discuss concerns related to the managing of projects. Therefore, the sparring activity is mainly aimed at less experienced project managers, but it also benefits from the possibility of identifying and discussing ideas for improvements related to the management of projects. However, it should be noticed that the sparring session is not a control mechanism used for monitoring the work of project managers. The role of the P^+ member is rather regarded as passive and listening as opposed to snooping and demanding. The project manager should be comfortable enough to discuss any concern, however large or small, with their sparring partner and take advantage of the P^+ member's experience and knowledge.

Startup Seminars

The goal of the *startup seminar* is to make sure that all projects start off on the right track. The startup seminar is meant as an alternative to the mandatory project kickoff prescribed by "Ratten". Typically, it will cover the internal part of the kickoff. The external parts concerning the customer must still be addressed separately. The startup seminar is carried out by so-called process consultants who could be members of P^+. Because of the process consultants, the project manager takes part of the seminar as a regular project member. Furthermore, all members of the project group must participate in the seminar taking place across one and a half day. The location must be external to the company, meaning that none of WM-data's conference rooms are suitable. It is emphasized that all project members must be present at all times during the seminar, both physically and mentally. This goal is only obtained if the seminar is held externally [Madsen, 2002].

The reason for conducting startup seminars is that almost all failed projects have gone off track during the startup phase, before the initiating the actual development process. This has been established during sparring sessions with members of P^+, as problematic content of such sessions often concern difficulties in setting up the project. The goals of the startup seminar are the following, as translated from [Madsen, 2002, pp. 4]:

- Establishing shared understanding of project objective.
- Establishing shared commitment to the project task at hand.
- Getting to know one another. Typically, some of the project participants have not been working together before, so social activities are to be encouraged
- Specifying expectations of one another, i.e. work routines, habits, personal objectives of project participants etc.
- Organizing the project. Identify and man roles of the project.
- Reviewing the task, e.g. requirements specification or design, in order to identify some of the problematic aspects in the present task

The process consultants are given a menu from which they can choose a set of items to be addressed during the seminar. The menu consists of the following items translated from [Madsen, 2002]:

It is stressed that the agenda for the startup seminar need not be strictly followed. If issues arise that are regarded as more important than items on the agenda, these issues may be addressed instead.

- Specification of project objectives	- Estimation
- Cooperative partner analysis	- Planning activities
- Potential solutions	- Scheduling
- Technical platform and tools	- Cooperation
- Small course in "Ratten"	- Quality management
- Risk assessment	

Table 3.2: *The eleven items on the menu.*

Theme Seminars

The *theme seminars* are used to address specific issues or topics that may be relevant at any given point during a project. The seminars take the form of conventional teaching where a lecture is held based on a specific topic. Such a topic could for instance be test or estimation. Everyone interested in the topic may participate. In this way, participation in the theme seminars is voluntary. The lecturer may either be a person from inside of WM-data or an external guest lecturer who possesses particular knowledge on the subject. In this way, theme seminars may be used to inspire new initiatives and generate ideas for improvements to current practices.

3.3.2 Organizational Setting of P^+

The P^+ initiative has currently gone past its startup phase, but it is still primarily driven by grassroot interests even though it has become part of a newly deployed Quality Assessment Group (QAG, for short). By doing so, WM-data attempts to put organizational focus on the QAG and thereby P^+. Our previous work shows how P^+ lacks organizational commitment and anchoring [Hosbond and Ørtoft, 2003]. This concern is addressed by the addition of the QAG to the organizational structure as shown in Figure 3.5. Hereby, WM-data aims at enhancing the visibility of division-wide initiatives such as P^+. It is, however, important to notice that the QAG is termed "*a virtual staff function*". The reason is that the nature of the QAG must be consistent with the informal company culture prevalent at WM-data. It is noticed by our contact person that adding such a strictly defined staff function to the organizational structure would lead to resistance and mistrust among employees and accordingly compromise the informal company culture. The QAG takes responsibility for several cross-divisional initiatives such as implementing the "delivery model", human resources, the P^+ initiative and *ad hoc* initiatives. P^+ thereby receives the organizational anchoring that has been lacking up until January 1st 2003.

The introduction of the QAG has, however, resulted in a shift in focus of P^+. In the previous research period, P^+ was seen as a informal SPI group that could launch cross-project SPI initiatives. In this way, they would be able to address some of the problematic issues that surfaced during sparring sessions on a more general level.

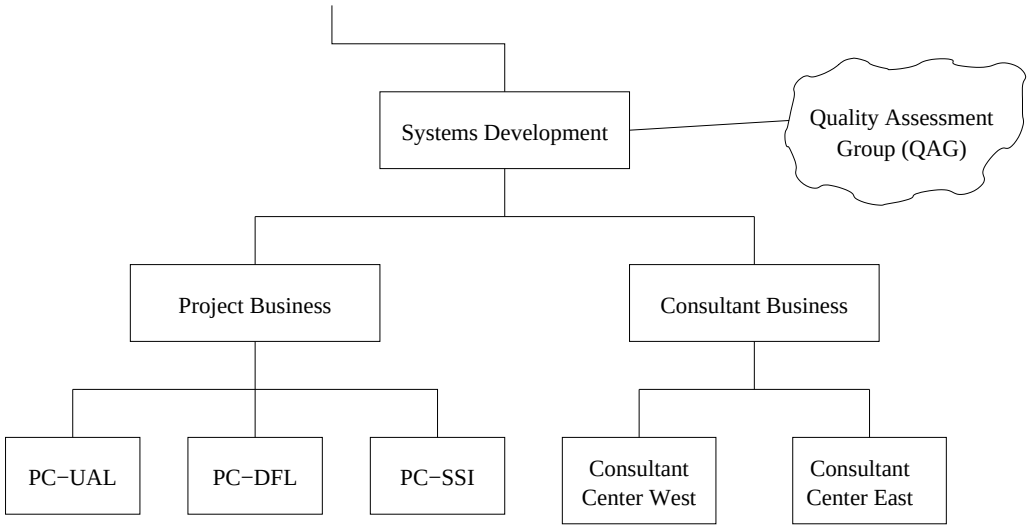


Figure 3.5: *The Systems Development branch of WM-data Denmark, as of January 2003.*

Such issues could then be debated on P^+ meetings, generating a shared understanding of potential solutions. Using their experiences, the P^+ members could then decide on how to handle the issues, either through division-wide improvement initiatives or through localized efforts.

The introduction of the QAG means that such improvement initiatives are no longer handled by P^+. Instead, they are managed by the QAG. Thus, the role of P^+ is rather to identify problematic aspects of WM-data's current software practices and decide on how to address the problematic aspects. However, the actual division of roles and their domain of responsibility are not clearly defined because many of the P^+ members are also members of the QAG. In this way, the work of P^+ and QAG may overlap, making an uneasy distinction between the two groups.

Also, since of January 1st 2003, WM-data's Systems Development branch has undergone some changes in its organizational structure. An attempt has been made to address the problematic East/West distinction by instead dividing the Project Center into three sub-divisions: PC-UAL, PC-DFL, and PC-SSI as depicted in Figure 3.5. Each of the sub-divisions focuses on one overall business area. PC-UAL is responsible for educational administrative solutions, PC-DFL centers on software for digital administration and PC-SSI focuses on health care and systems for embedding/integrating systems. The Aalborg department is located inside the PC-DFL which consists of 15 project managers equally distributed among three geographical locations: Aalborg, Aarhus, and Ballerup. Furthermore, group leaders have been introduced on each location such that a close contact between the systems developers and their group leaders is strengthened. However, the Consultants Center is still

divided into an East department and a West department.

In effect, P^+ may then be viewed as a group that accumulates experiences gained via the projects run in the Systems Development branch. P^+ is therefore very interesting from a knowledge management perspective. Further reasons for working with the P^+ initiative are given in the next subsection.

3.3.3 Why P^+?

When comparing the above aspects of P^+ to the company profile and the overall nature of the Systems Development branch, some important points can be made concerning ways of improving current practices. With respect to their daily practices, WM-data relies heavily on their very informal company culture in daily practices; members of the organization address each other directly when problems or concerns arise, taking short informal meetings in hallways and offices. This means that P^+ cannot formally manage knowledge, but must rely on informal practices.

Another point concerns WM-data's mixed history with large-scaled improvement initiatives. It seems that WM-data has difficulty establishing organization-wide commitment towards such initiatives. For instance, WM-data tried to get certified according to ISO 9000 during the eighties. This effort failed in spite of several attempts. Another example is the adoption of "Ratten" to all projects. "Ratten" was introduced in 1998, but has still not been fully implemented in all projects. One last example is WM-data's use of experience groups. Even though several groups have been established, they loose focus and momentum because members tend to focus more on doing their job than tending to knowledge sharing via the groups. In effect, knowledge sharing is acknowledged, but time is not set aside for knowledge sharing activities. All of these examples suggest that WM-data lacks organizational focus with respect to initiatives which do not directly involve a paying customer. Instead, WM-data encourages people only to concentrate their efforts on systems development and running projects by not formally appreciating knowledge sharing activities. Thereby, only short-term goals of development are fulfilled. In effect, P^+ suffers a great deal from lack of resources because upper management is not willing to commit to knowledge sharing activities. In this way, the long-term goal of software process improvement is compromised, favouring the short-term goal of running profitable projects. The previous failed attempts and the problems experienced with the implementation of organization-wide initiatives indicate that our area of focus should be focused enough to maintain the commitment of stakeholders.

Another point to be noticed is the fact that while WM-data is well-established with respect to technical knowledge and customer-related knowledge, their process-related knowledge is at the core knowledge level [Hosbond and Ørtoft, 2003]. The projects at WM-data are run almost independently and only little knowledge established during projects is (re)used by other projects. The P^+ initiative may potentially affect this situation by acting as the locus for cross-project initiatives. In this respect, P^+ may

be viewed as a SPI unit affecting projects via the project managers. Furthermore, it seems as if P^+ receives the commitment that the formerly mentioned cross-project initiatives lacked, in spite of the fact that P^+ ideas have been around since 2000. Also, P^+ is run rather informally without a formal set of rules or guidelines. Through the interviews of the previous research period, all project managers expressed content with the underlying idea of sparring.

P^+ may therefore show to be an initiative that upper management at WM-data is willing to support. Even if the initiative is grassroot-driven, it has been well-acknowledged by key individuals in the organization. Because of the above argumentation and, in our opinion, obvious benefits of P^+, we have chosen to focus on the P^+ initiative and investigate which improvements can be suggested from a knowledge management perspective to support the work of P^+. In effect, this may increase P^+'s chances of being properly supported by upper management and thereby increase the chance of becoming organizationally anchored. Thus, our motivation for cooperating with P^+ is the following question:

> *What can be done from a knowledge management perspective to support the work of P^+?*

This question directs the SSM analysis documented in the next chapter.

Chapter 4

SSM Analysis

In this chapter, we conduct an SSM analysis with focus on P^+. Before suggesting systems to improve or support the problematic situations of P^+, we organize our background knowledge about P^+ and their surrounding environment by making the cultural enquiry of SSM. Understanding the characteristics of P^+ and the context in which they exist, we continue with the logic-based enquiry. We conclude the SSM analysis by discussing the results of the comparison stage which leads to the selection of relevant systems to be experimented with during prototyping.

Our logic-based enquiry has been conducted through four iterations of the seven stage model, as depicted in Figure 2.4. Figure 4.1 shows an overview of the four iterations with a short description of how each stage has been adopted. This should help to understand the focus of each iteration and how the iterations interrelate. We now turn to the cultural enquiry.

4.1 Cultural Enquiry

The cultural analysis of WM-data in this study has been made continually, but the basic understanding of roles, the social system and the political system has been gained through the case study conducted prior to this research period, see [Hosbond and Ørtoft, 2003]. It is, however, useful also to summarize these findings according to Checkland's three analyses of the cultural enquiry as described in Subsection 2.3.4.

4.1.1 Analysis One: Roles

Our cooperation with WM-data was initiated due to a request from WM-data. Initially, they had a rather vague idea about working with issues related to knowledge management, but they expressed great interest in the subject. This meant that we,

Stages	1. Iteration	2. Iteration	3. Iteration	4. Iteration
1.	Finding out. Applying background knowledge from the cultural enquiry	–	–	–
2.	Expressing the problematic situation through EROS	Expressing various perspectives on the system through a rich picture	Taking a KM per− spective. Personali− zation (p) vs. codifi− cation (c). Leads to 6 system ideas	–
3.	Defining 3 systems all applicable to EROS	Formulating 7 different system ideas based on the rich picture	Formulating root definitions and CATWOEs/CAIWOEs of the 6 systems	–
4.	The EROS model	Rich picture	Building 6 conceptual models	–
5.	Comparison of 3 systems	Comparison of 7 systems	Comparison of 6 systems	Comparison of 4 systems. Conducted with P+ member
6.	Selecting 1 system	Selecting 3 systems (S1, S4, S5)	Selecting 4 systems (1c, 4c, 5p, 5c)	Selecting 2 systems (1c, 4c)
7.	Initiate 2. iteration	Initiate 3. iteration	Initiate 4. iteration	Initiate prototyping based on the agenda system (1c) and the war stories system (4c)

Figure 4.1: *Overview of the SSM analysis.*

through a joint venture, could ensure that both parties (WM-data and ourselves) would benefit from the cooperation.

Clients

In the current research period, our focus has been on P^+ in the Systems Development branch of WM-data. Following Analysis One in the cultural enquiry, explained in Subsection 2.3.4, the role of our clients is thus the members of P^+.

Problem Owners

With respect to the role of problem owners, it found that they can be divided into a group of primary problem owners and a group of secondary problem owners. The primary group of problem owners is the group constituted by the P^+ members. Obviously, through intervention, we try to support and improve problematic aspects of the current practices of P^+. Seen on a broader scale, we argue that the entire branch of Systems Development constitutes the secondary group of problem owners. The argumentation is straightforward. Having the purpose of P^+ in mind, namely to improve project management and projects in general through sharing of project experience, we argue that the problematic aspects of P^+ restrict the quality of the improvement efforts initiated by P^+, which affects the entire branch of Systems Development.

Problem Solvers

The role of problem solvers is assumed by both our cooperative partner at WM-data and ourselves. Through the process of our study and in particular through our process of SSM and prototyping, we interact with a highly experienced and respected project manager and member of P^+. By analyzing problematic aspects of P^+ through the use of SSM, we seek to understand the requirements of the client, P^+, and the organizational context in which P^+ exists. This means that we have initiated several debates and discussions, not only in the early phases of our analysis, but continually as our analysis and prototyping activities have progressed. Consequently, through our process of intervention and through the feedback delivered by the project manager at WM-data, both parties become problem solvers.

4.1.2 Analysis Two: The Social System

Analysis Two covers a mapping of roles in the social context of P^+ and their relation to the norms and values in the Systems Development branch. As explained in Section 2.3.4, norms denote the expected behaviour of a role, whereas values represent the local standards used for judging whether the actual behaviours of a role are acceptable or not. Like in Analysis One, we limit the investigation of Analysis Two to the area representing our research focus, namely P^+. By initiating Analysis Two, we hope to reveal any problematic aspects of P^+ related to social problems of the environment surrounding P^+. We start off by describing the different roles prevalent in P^+.

Roles

The roles in P^+ consist of both formal and informal roles. The formal roles relate to the social position of P^+ members and persons who may affect the work of P^+. The social position of P^+ members are the position of project managers. Other parties that may affect the work of P^+ are project managers, senior managers, members of the QAG and systems developers. On the other hand, the informal roles become apparent when P^+ members participate, initiate and conduct activities aimed at sharing project experience, e.g. a P^+ member involved in a sparring session. This leads to the identification of the following informal roles which may be assumed by P^+ members:

- Share and discuss project experiences on P^+ meetings
- Evaluate current P^+ activities
- Conduct sparring sessions with project managers not part of P^+
- Plan and conduct startup seminars
- Plan and conduct theme seminars

Secondary, yet plausible informal roles are:

- Discuss project experiences and P^+ initiatives with senior manager
- Revise the process model, "Ratten"
- Coordinate initiatives with the QAG

Norms

For each role there exists a set of expected behaviours created by the surrounding environment, i.e. employees, customers and company culture. Having defined formal roles such as P^+ members, project managers, systems developers, senior managers etc., we might go into a deeper elaboration on the expectations linked to each role. However, since the focus of this research has been directed towards a thorough understanding and analysis of P^+, we will only consider norms related to the formal roles of the P^+ members. These norms are seen from the eyes of the other formal roles earlier presented. The following elaborations are taken from our previous study in which we interviewed three groups of people from the Systems Development branch, namely systems developers, project managers and one senior manager. Through the interviews, we came to understand the expectations that each group had for the work of P^+ and therefore also for the members of P^+. The most dominating expected behaviours are summarized by the following set of keywords:

- Non-controlling management style
- Human understanding
- Diplomacy
- Democracy
- Listening

Whether or not these keywords correspond to the actual behaviour are often judged according to some local standards of correct or incorrect behaviour also referred to as values, Checkland and Scholes [Checkland and Scholes, 1990]. These values are elaborated on next.

Values

The values of P^+ members strongly relate to the values belonging to project managers in general. We argue that P^+ has unconsciously adopted the set of values belonging to the group of project managers. Several reasons exist. For one, P^+ is still a grassroot- driven initiative, meaning that the initiative is based on personal beliefs from dedicated and experienced project managers. The fact that the project managers are experienced and have a long history within WM-data means that all P^+ members have adapted to the informal company culture and probably appreciates it (after all, if the opposite was the case they would most likely have looked for another job). A second reason is the fact that P^+ has just recently begun conducting sparring sessions which limits the amount of experiences from this activity that could affect the set of values. Furthermore, the sparring sessions concern counseling of project managers. It would be obvious to maintain a set of values similar to those of project managers in order to foster a shared context as proposed by Nonaka and Konno [Nonaka and Konno, 1998]. We therefore argue that these issues together affect the values of P^+ and that these values are the same as the values for project managers.

To confirm this argument, we have found that people engaged in activities involving a P^+ consultant all express great satisfaction and content towards both the initiatives and the way the initiatives are conducted. Even though this shows that the early experiences related to the involvement of P^+ have been positive, we still believe that the activities of P^+ have to be further incorporated into current practices before concluding whether or not the work of P^+ will be fully accepted according to the norms and values of P^+.

4.1.3 Analysis Three: The Political System

The political system at WM-data is very much affected by the project culture dominating practices in the Systems Development branch. Through the qualitative interviews conducted in our previous study, interviewees emphasized that it is the projects that keep the great company wheel turning. This means that the projects have a huge impact on the political system. Furthermore, the project culture has been very much affected by the informal company culture. In this respect, P^+ can be regarded as just another group within the set of projects. Even though, the purpose of P^+ may differ in nature from the usual goals of traditional projects, it is, however, still a community of people affected by the existing project culture. We therefore argue that the political system in the Systems Development branch also includes P^+. In Subsection 2.3.4, we presented three questions which Checkland and Scholes have used to describe the political system [Checkland and Scholes, 1990]. For convenience, we restate them here:

- *What are the commodities through which power is expressed?*
- *How are these commodities obtained, used, protected, preserved, passed on, relinquished?*
- *Through what mechanisms?*

The answers to these questions must be sought through our observations, discussions and qualitative interviews conducted in our previous study [Hosbond and Ørtoft, 2003]. Before answering the three questions, we find it important to explain the subject *power* in the sense of the project culture at WM-data.

As stated in our previous study, we found that the Systems Development branch of WM-data is characterized by an informal company culture. More specifically, this is illustrated in the informal communication, the willingness towards helping others and the resistance towards constructing or adding new formal organizational structures and authorities. The downside of this latter fact has earlier been elaborated on, namely that P^+ has experienced severe trouble in getting properly anchored in the organizational context. Consequently, having the informal company culture and the project culture in mind, we argue that power only to some extent depends on the job title, but rather on a project manager's ability to e.g. communicate, help co-workers, engage in new initiatives, his social nature, quality of his work and his ability to manage projects within the boundaries of time and economics. This concludes the subject power in relation to the project culture and also answers the first question of the power-related questions listed above. Based on our elaboration of power the answers to the second and third question also become obvious. Power is obtained, used, protected etc. through social skills, level of knowledge and the ability to deliver good results. In addition, these issues also represent the mechanisms through which power is expressed.

4.1.4 Analysis Four: Knowledge

The previous three analyses constitute the cultural enquiry presented by Checkland and Scholes [Checkland and Scholes, 1990]. However, due to our focus on the management of knowledge in WM-data, we propose a fourth dimension to the cultural enquiry, namely that of Analysis Four concerning knowledge. Through our previous study, we investigated different types of knowledge, knowledge levels and how knowledge in general was managed in WM-data [Hosbond and Ørtoft, 2003]. Our findings were compared to relevant theory within knowledge management. Based on our empirical investigation of knowledge management at WM-data, we formed a company profile, also described in Subsection 3.2.2. The company profile presents an overall picture of the characteristics constituting current practices at WM-data. Furthermore, it indirectly reveals how knowledge is currently managed. However, with respect to issues related to knowledge management we emphasize the following keywords from the company profile:

- Personal networks
- Informal communication
- Grassroot SPI efforts
- Isolated projects
- Customized products

Based on these company characteristics, we now informally describe the current knowledge strategy at WM-data and in P^+ as such. The knowledge strategy at WM-data is very much informal, implying that WM-data has not yet formulated a knowledge strategy. However, we argue that WM-data possess an informal knowledge strategy formed by the company culture. The company culture is dominated by informal communication, openness and willingness to help and support co-workers. Due to the missing formal knowledge strategy, there exist no formalized activities with the sole purpose of sharing, capturing or creating knowledge. The process of creating and sharing knowledge is therefore integrated as part of daily practices, e.g. co-workers share knowledge and experiences when initiating informal communication. Consequently, the creation or sharing of knowledge takes place more or less spontaneously which in the words of Mathiassen et al. may result in the creation of situated knowledge [Mathiassen et al., 2003]. Following Zack, WM-data is forced from time to time to create situated knowledge which is also innovative in order to stay competitive [Zack, 1999] and deliver customized products. In this respect, the current informal knowledge strategy satisfies the knowledge needs caused by product demands. However, the usefulness of the informal knowledge sharing is limited when projects are geographically separated as is the case in WM-data. This has shown to be one of greatest limitations of the current knowledge strategy.

Another problematic aspect is the lack of formalized processes in which knowledge is managed. We argue that the knowledge strategy must be a combination of formal and informal activities in order to fulfill the continuous demands of innovative knowledge at WM-data. Formal activities must, however, not compromise the informal practices of WM-data and P^+ in particular. Instead, formal activities should support and improve existing practices. But, if formal activities are required, to what extent should formal activities be based on codification? We believe this to be the main question which needs to be answered in order to implement a knowledge strategy that fits the existing company culture. Hansen et al. propose two complementary knowledge strategies, namely the personalization strategy and the codification strategy and argues that a knowledge strategy should be based on a combination of both the personalization strategy and the codification strategy [Hansen et al., 1999].

During the logic-based enquiry of SSM as initiated in the following sections, we intend to propose and analyze systems with respect to these two perspectives: Personalization and codification.

4.2 1st Iteration: Finding Out

With the outset of the motivating question from the problem setting:

> *What can be done from a knowledge management perspective to support the work of P^+?*

we start off our SSM analysis with a focused view on the initial problem situation: P^+ from a knowledge management perspective. Consequently, it must be stressed that our mission is not to change the basic nature of P^+ or to rethink or change the organizational role of P^+. Hence, we will not consider the problematic aspect of anchoring P^+ in the organizational context of WM-data as described in Section 3.3. Instead, we intend to support existing work practices within P^+, i.e. activities within P^+ that are highly dependent on how knowledge is managed. In this light, we may recommend certain courses of actions that may improve particular aspects of P^+, including their four main activities: P^+ meetings, sparring sessions, startup seminars, and theme seminars.

However, as pointed out by Zack, knowledge management must be adopted so that it fits the overall knowledge strategy of the company [Zack, 1999]. This is, however, a difficult objective to obtain in the case of WM-data simply because a formalized knowledge strategy does not exist in explicit form; it is embedded as part of the overall goals, visions and the informal company culture of WM-data, as described in Section 4.1. Therefore, proposed knowledge management approaches correspond to WM-data's informal way of managing knowledge.

4.2.1 Expressing The Situation P^+

Having stressed the focus of our analysis, we now analyze how we can actually support and improve the knowledge dependent activities within P^+. To get a notion on how P^+ works, or at least is intended to work, we adapt the EROS model (Environment, Relations, Operations and Systems) as applied by Checkland and Scholes [Checkland and Scholes, 1990] instead of utilizing a rich picture. The reason for using the EROS model as a pictorial representation of P^+ instead of a rich picture is that it is easier to express P^+'s situation via EROS because of the knowledge gained through our previous study. Such a focused view on P^+ is difficult to express in a rich picture because such a representation rather serves as an expression of problematic aspects. We know this because we have tried it. Figure 4.2 thus shows an EROS model illustrating P^+'s role in WM-data conforming to the main objective or mission of P^+: Improving project management through sharing experiences and project related knowledge between projects.

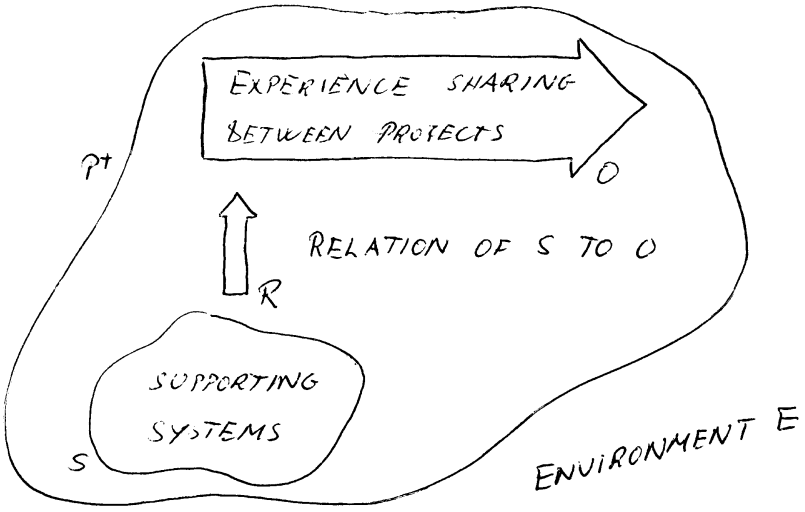


Figure 4.2: *The main objective underlying P^+ modelled based on EROS [Checkland and Scholes, 1990].*

Figure 4.2 shows how the main operation of P^+, O: 'Experience sharing between projects' is enabled by a set of supporting systems, S. The relationship between O and S is depicted as R. This relationship symbolizes the importance of tailoring S to fit properties characterizing O. All of this takes place in the environment, E, which constitutes the organizational context.

From the EROS model of P^+'s role in WM-data, a number of relevant systems arise:

1. A system for management of knowledge aspects involved in the work of P^+.
2. A system for continual evaluation of R.
3. An acceptance system for alleviating potential conflicts between P^+ and its organizational environment, E, aimed at increasing the anchoring of P^+ in its environment.

The first system concerns the management of knowledge in its broadest sense. Consequently, the system covers all subsystems related to the management of knowledge, i.e. acquisition, sharing, use, capturing and creation of knowledge following Kautz and Thaysen [Kautz and Thaysen, 2001]. The subsystems may cover the whole spectra from a personalization strategy to a codification strategy.

The second system represents the perspective of continual evaluation of initiatives involved in the work of P^+. Such evaluation could be aimed at answering questions such as: Is the supporting systems doing what they are supposed to do? Does a certain initiative address the correct concerns or has there been a shift in focus? and Could other supporting systems be of relevance? Such a system should be an integrated part of P^+'s work.

The third system addresses conflicts that might exist between P^+ and its environment. The work of P^+ may potentially conflict with various parts of WM-data. Such conflicts could for instance arise between P^+ and project managers or between P^+ and senior management.

Of the three systems, we choose to focus our further analysis on the first system. The reason for not choosing the second system is that such continual evaluation would be hard for us as students to contribute to because the system presupposes the existence of supporting systems. Currently, the supporting systems for P^+'s main operations are their four main initiatives: P^+ meetings, sparring sessions, startup seminars and theme seminars (see Subsection 3.3.1). Taking the informal nature of P^+ into consideration, such initiatives are best evaluated via informal discussion and they seem reluctant to formalize this process. With respect to the third system, it would be unrealistic to think that we as students could contribute to the acceptance of P^+ in the organizational context of WM-data. Rather, we may *suggest* improvements to be made which WM-data may take into consideration or not. Therefore, we choose to focus our efforts on the first system for managing knowledge aspects involved in the work of P^+, making the first system the starting point of a second iteration of SSM.

4.3 2nd Iteration: Proposing Systems

The second iteration aims a proposing relevant systems to support or improve existing knowledge dependent activities within P^+. This means that the second iteration

is only concerned with aspects of P^+ that are relevant seen from a knowledge management perspective. This narrowed focus leads to a new initial problematic situation (SSM stage 1) expressed in the rich picture shown in Figure 4.3.

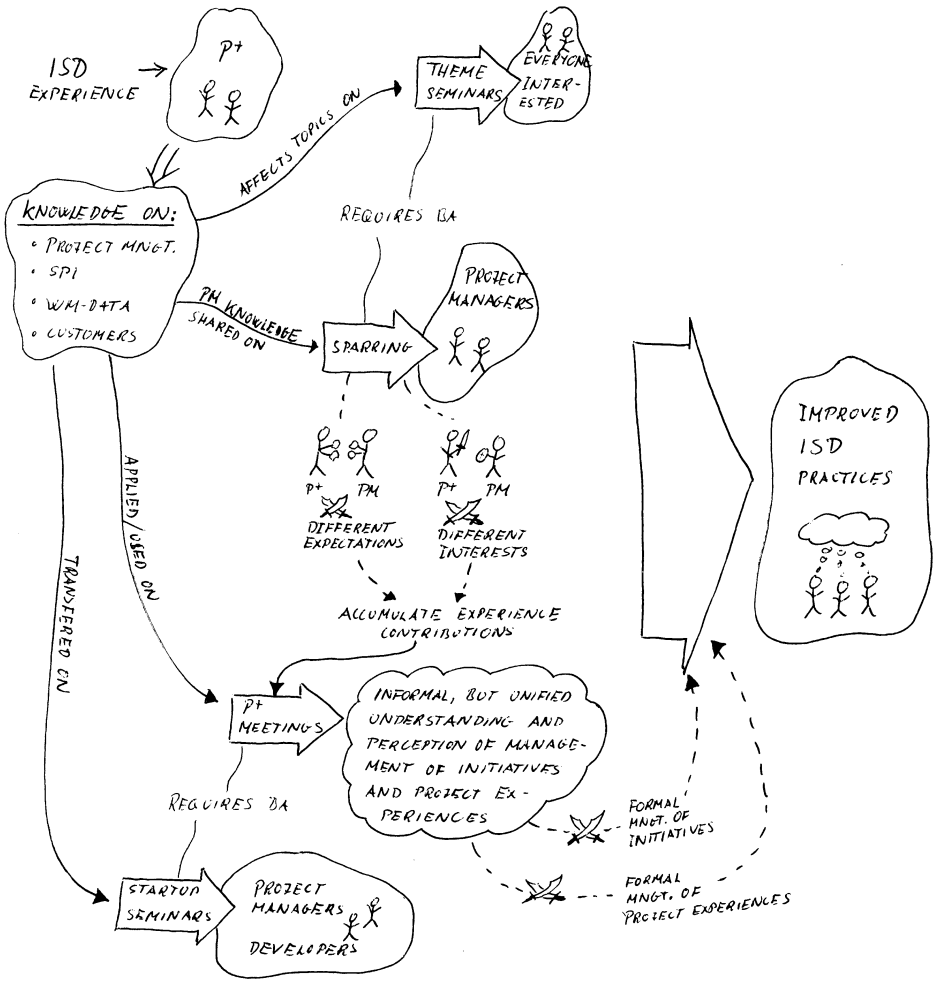


Figure 4.3: *A rich picture describing P^+'s four main means of acquiring, sharing, capturing, using, and creating project knowledge.*

Figure 4.3 shows the four main activities of P^+, namely theme seminars, sparring sessions, P^+ meetings and startup seminars. These activities are all characterized by either one or more of Kautz and Thaysen's basic five aspects of knowledge management: capturing, sharing, creating, acquiring and using knowledge [Kautz and Thaysen, 2001].

The rich picture of Figure 4.3 has lead to the identification of seven relevant systems for improving and supporting at least one of the four main activities presented in

Figure 4.3:

S1: Provision, acquisition and sharing of project experience through discussions and evaluations on P^+ meetings and sparring sessions.

S2: Provision of resources to conduct P^+ meetings and initiate P^+ initiatives.

S3: Coordination of P^+ members' usage of project experiences on sparring sessions.

S4: Fostering commitment and engagement for participation in P^+ meetings and sparring sessions.

S5: Management of P^+ initiatives.

S6: Coordination of P^+ initiatives with the QAG.

S7: Obtain organizational access to project experience managed in P^+.

From the seven systems, it becomes apparent that the degree to which each system relates to knowledge activities differ. Moreover, the degree to which we as students may contribute to their success seems minimal. The reason is that some of the systems relate to improving internal political affairs in WM-data, also referred to as *power struggles* in terms of Analysis Three of the cultural enquiry. Such change seems unrealistic having our current institutional position as students in mind. Of the seven systems, we therefore focus our third iteration on the following the three systems:

S1: Provision, acquisition and sharing of project experience through discussions and evaluations on P^+ meetings and sparring sessions.

S4: Fostering commitment and engagement for participation in P^+ meetings and sparring sessions.

S5: Management of P^+ initiatives.

The remaining four systems (S2, S3, S6 and S7) may be just as important, but they represent areas which cannot solely be dealt with by only focusing on P^+. Taking our restricted amount of time into consideration, we therefore continue with the third iteration of our SSM analysis, focusing on the three selected systems: S1, S4 and S5.

4.4 3rd Iteration: System Perspectives

In the third iteration of SSM, we focus on viewing the three systems, singled out in the previous iteration, from different knowledge management perspectives. In this

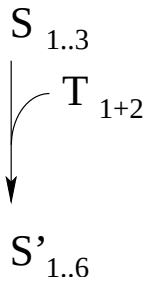


Figure 4.4: *The process of generating new systems.*

way, knowledge management is used to suggest six new systems, providing different perspectives on the three generic systems. This process is visualized in Figure 4.4.

Six systems, $S'_{1..6}$, are generated based on the three generic systems, $S_{1..3}$, and two overall knowledge management perspectives, T_{1+2}. The overall knowledge management perspectives used in this generation are the codification strategy and the personalization strategy suggested by Hansen et al. [Hansen et al., 1999], briefly described in Subsection 3.1.4. The reason for choosing these two perspectives is that in Analysis Four, Subsection 4.1.4, we emphasized the need for a formalized knowledge strategy guiding the management of knowledge. Therefore, we view the three generic systems (S1, S4 and S5) from the perspectives of both personalization and codification. In addition, these two strategies are coupled with other aspects of knowledge management theory in order to suggest new systems.

We now turn to the presentation of six systems which serve as perspectives on the three selected systems from the second iteration of SSM, see Section 4.3. Each presentation of the different perspectives includes a reference to the generic system, a description of the chosen knowledge management strategy (personalization/codification) and a presentation of the new system, including root definition, CATWOE / CAIWOE and conceptual model. As suggested by the terms CATWOE and CAIWOE, we use both transformation systems and interaction systems in forming conceptual models. Furthermore, root definitions and CATWOEs / CAIWOEs are meant to supplement each other, i.e. the root definitions do not necessarily reflect what has been stated in the CATWOEs / CAIWOEs. As a general remark to the CATWOEs and CAIWOEs, the 'customers' only constitutes primary beneficiaries. Secondary beneficiaries such as WM-data as a whole are not considered.

Another point concerns the use of the three Es (*E*fficacy, *E*fficiency and *E*ffectiveness), see Subsection 2.3.3. We have found these activities to be very loosely handled by P^+. There has never been a formal way of monitoring or judging whether initiatives were conducted as expected (efficacy), had the expected outcome compared to the amount of resources spent (efficiency) or satisfied longer term goals (effectiveness). Rather, it has been based on involved people's informal perception of

the initiatives. In addition, as long as it is grass-root driven improvement efforts that represent the improvement efforts in WM-data, we find it unrealistic that P^+ would make use of activities similar to those representing the three Es simply due to the lack of resources and due to traditional way of evaluating such systems, namely through informal discussions. We therefore adapt P^+'s informal way of evaluating the system with respect to the three Es, by only informally taking them into consideration.

4.4.1 System S1

In this Subsection, we present different perspectives on the the first generic system, see Section 4.3. For convenience, we restate the system here:

S1: Provision, acquisition and sharing of project experience through discussions and evaluations on P^+ meetings and sparring sessions.

The purpose of this system is to provide means of sharing and acquiring project experiences to project managers. In this subsection, we present two perspectives on this system

The Broker System (1p)

By taking the perspective of a personalization strategy combined with an idea adapted from Hellström et al. [Hellström et al., 2001], a new system emerges termed *broker system*. Through this new system, project experiences may be shared and acquired through personal interaction between actors facilitated by *knowledge brokers*. Following Hellström et al. [Hellström et al., 2001], the basic idea is that P^+ members either full-time or part-time should act as knowledge brokers who continuously keep track of knowledgeable people (experts) and their specific areas of expertise. The knowledge brokers then establish contact to these sources when someone in the organization needs help. Knowledge brokers are supposed to take on a demand-driven approach in the words of Swan [Swan et al., 1999], establishing contact when knowledge is requested. By doing so, the aim is to share knowledge between projects.

The broker system is modelled as an interaction system because it concerns the management of information rather than a transformation with a well-defined input and output. The CAIWOE and root definition of the broker system are formulated as follows:

CAIWOE: Broker system (1p):

C: Members of P^+ and project managers.

A: Members of P^+ (knowledge brokers).

I: Managing contact information through mapping between knowledge experts and knowledge seekers.

W: By applying a pure personalization strategy, the informal company culture is supported and it is ensured that knowledge requests are handled through personal contact with experts, increasing the rate of learning and understanding.

O: P^+.

E: Knowledge brokers must be available and be able to handle requests immediately such that knowledge seekers do not have to wait too long for the personal contact.

Root definition: Broker system (1p):
A system in which one or more skilled, experienced and respected employees (knowledge brokers) link knowledge seekers to relevant experts by maintaining a large personal network and by keeping up-to-date on the latest concerns or problematic areas concerned with project management, in order to increase the communication flow of project-related knowledge between projects.

Based on the root definition, CAIWOE and the basic description as such, the conceptual model of Figure 4.5 has been formed.

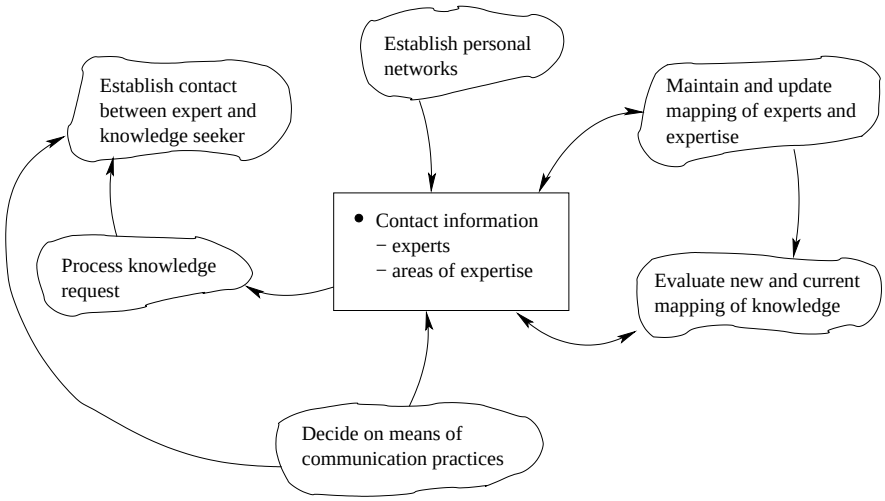


Figure 4.5: *Conceptual model of the broker system (1p).*

War Stories System (1c)

Taking a codification perspective on S1 suggests for explication of knowledge. One way of sharing project experiences is through storytelling which is widely used in communities-of-practices [Brown and Duguid, 1991]. Brown and Duguid refer to a particularly emotional version of such stories called *war stories*. Stories categorized as war stories represent personal experience that have made a deep emotional impact on the storyteller. Such experiences may either be in the form of best practices or worst practices — anything which moves the storyteller. Another aspect is that such war stories may be retold by others who have also been moved by their contents.

The basic idea is that project experiences may be shared through the telling of such war stories. With this in mind, a new system may be proposed that incorporates both the concept of a war story and takes on a codification strategy. The key belief is that war stories are capable of enriching the articulation of experiences or knowledge such that both are more efficiently shared and clearly understood.

War stories are either in the shape of general experiences or examples of best or worst practices. They can be applied when a project manager seeks knowledge on what course of action should be chosen when encountering a problematic situation. Experiences from actual practices may be shared between P^+ members and project managers during sparring sessions. The experiences may be discussed and systematized into war stories via the new system which appears when viewing S1 from the perspective of storytelling. Tacit knowledge in the form of stories from actual practices is converted into codified war stories without loosing the context which enriches the perception of the story.

The war stories system is modelled as a transformation system. The CATWOE and the root definition of the war stories system are formed as follows:

CATWOE: War stories system (1c):

C: Project managers and P^+ members.

A: P^+ members.

T: Personal tacit work-related experiences $\rightarrow$ codified and shared war stories.

W: Personal experiences are very valuable to both individuals and the organization. They are, however, difficult to share. War stories are a solution where such experiences concern both best and worst practices. Through the telling of documented war stories, the teller may be able to make a deeper impact on the listener than by normal conversation because the war stories typically represent emotional tales. This influences the perception of the story.

O: P^+.

E: The war stories must conform to a prescribed set of standards so that they are uniformly structured. The format must ease the application of codified stories.

Root definition: War stories system (1c):
A system that organizes work-related experiences, by means of codifying, sharing and evaluating war stories, in order to obtain a common understanding of best and worst practices.

Again, we infer the conceptual model of the new system from the root definition, CATWOE and the description above, see Figure 4.6.

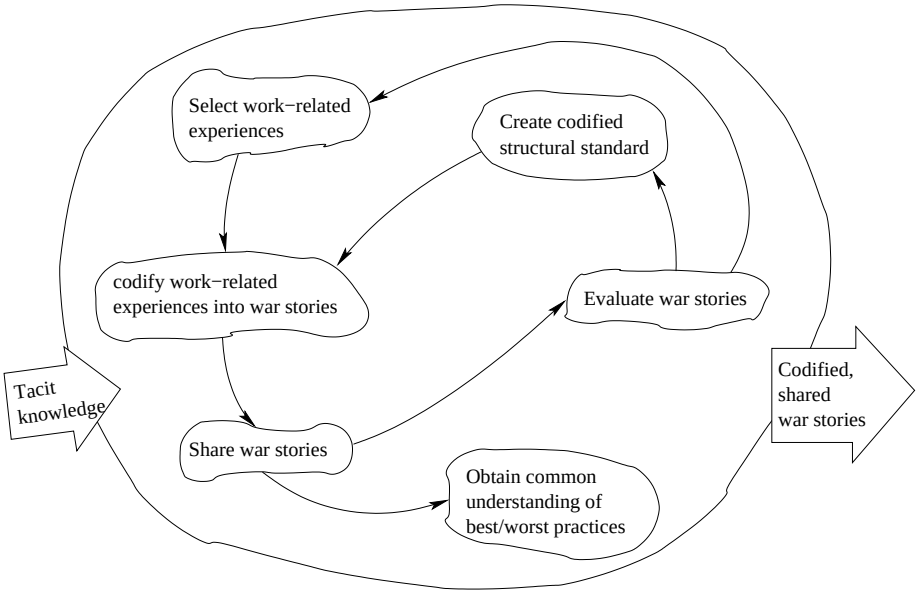


Figure 4.6: *Conceptual model of the war stories system (1c).*

4.4.2 System S4

In this subsection, we present two different perspectives on the fourth system:

S4: Fostering commitment and engagement for participation in P^+ meetings and sparring sessions.

Reward System (4p)

By taking on a personalization perspective, the fostering of commitment in S4 can be done by rewarding participation in P^+ initiatives.

The basic idea is that each employee (project manager or P^+ member) has a personal list of contributions that represents their engagement and willingness to participate in improvement activities such as sparring or P^+ meetings. The list is maintained and updated by P^+. The intent is that the list can help the project manager or P^+ member when he/she applies for another job, either internally or externally.

The system is modelled as an interaction system because it concerns the management of organizational commitment. The CAIWOE and root definition are formulated as follows:

CAIWOE: Reward system (4p):

C: Project managers and P^+ members.

A: P^+ members, project managers and individuals.

I: Fostering of commitment managed through documenting initiatives and contributions of each project manager or P^+ member.

W: Commitment may be fostered organizationally by supervising the contributions and initiatives made by each project manager or P^+ member. Rewarding project managers and P^+ members for their willingness and interest in contributing to the overall improvement efforts conducted by P^+, may increase the organizational anchoring of P^+ and motivate others to participate.

O: P^+.

E: The individual or group that maintains the contribution lists must not be affected by personal relations, compromising the objective evaluation of contributions.

Root definition: Reward system (4p):

A system for rewarding P^+ members and project managers who contribute to the informal base of project knowledge and engage in P^+ meetings or sparring sessions, by documenting the initiatives and contributions of each project manager or P^+ member for use in later career-related situations, in order to motivate and increase interest and commitment towards participation in P^+ initiatives.

The conceptual model of the new system is depicted in Figure 4.7.

Agenda System (4c)

By taking on a codification perspective on S4 and combining this perspective with the idea of shared agendas for fostering commitment, a new system emerges aimed at

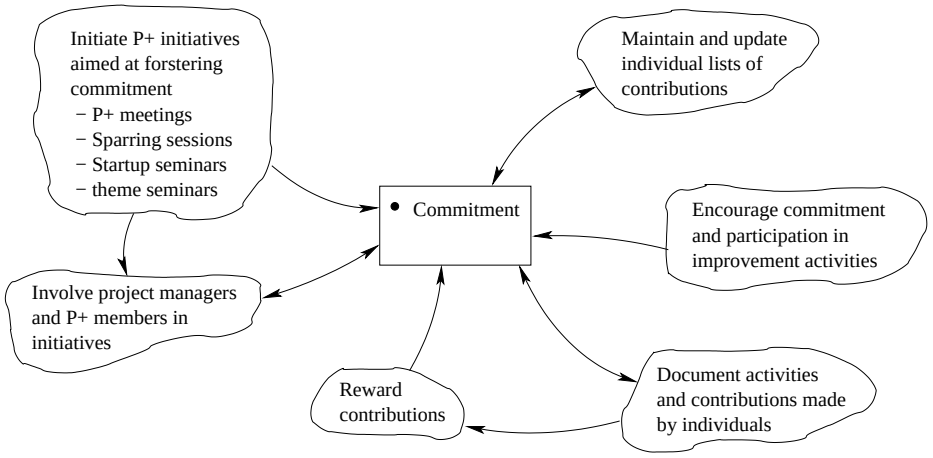


Figure 4.7: *Conceptual model of the reward system (4p).*

supporting an actual P^+ activity, namely that of P^+ meetings. Such shared agendas may furthermore help P^+ members form *bas* on P^+ meetings.

The basic idea of the agenda system is that all participants should be able to contribute to the agenda, thereby forming the topics for the next P^+ meeting. Important topics may be proposed, commented on by peers and discussed among group members. The goal is to make way for establishing a shared context prior to the actual meeting, that is, to create a shared understanding and commitment to the topics which are to be addressed on the meeting. Moreover, a P^+ member has to be responsible for making the final agenda for a meeting based on proposed topics.

The CATWOE and root definition for the new system are as follows:

CATWOE: Agenda system (4c):

C: Members of P^+.

A: Members of P^+.

T: Weak ideas for agenda $\rightarrow$ Shared agenda.

W: Through a shared agenda, participants are more likely to be focused, committed and motivated when meetings are initiated. The outcome of the meeting will likely increase if a shared context and commitment are established.

O: P^+.

E: Administrative overload, lack of commitment.

Root definition: Agenda system (4c):
A system to manage the creation of group agendas, by systematizing and

prioritizing agenda items that may be discussed and commented by participants, in order to organize meetings and foster both commitment and the creation of a shared context for all participants.

The conceptual model for the agenda system is seen in Figure 4.8.

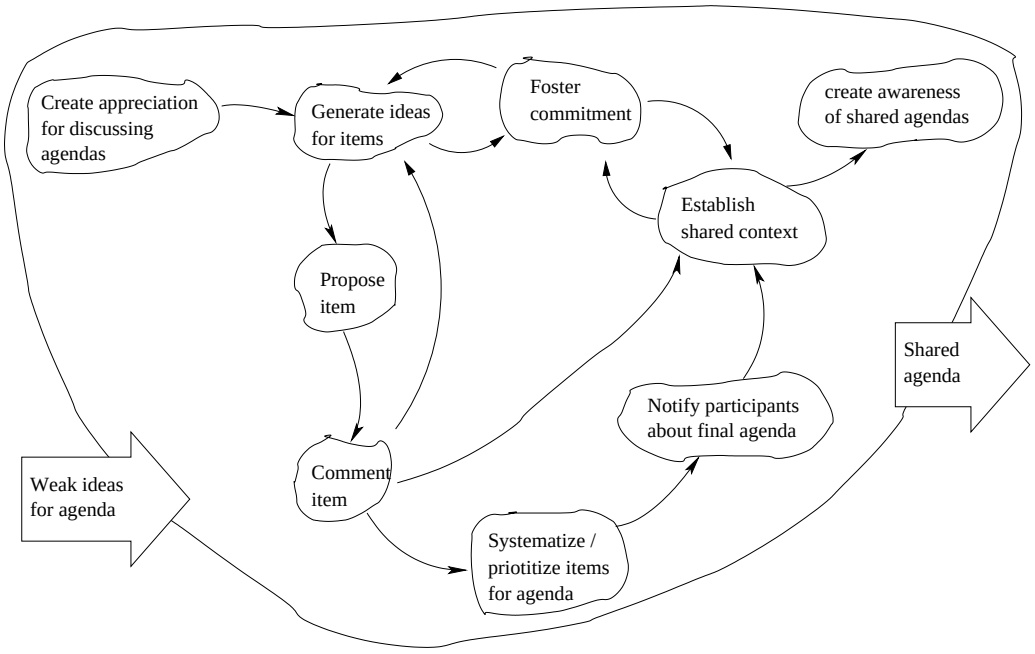


Figure 4.8: *Conceptual model of the agenda system (4c).*

4.4.3 System S5

The fifth system relates to the management of P^+ initiatives:

S5: Management of P^+ initiatives.

The purpose of this system is straightforward. Only by evaluating existing initiatives can these be improved in order to ensure the optimal outcome. In this Subsection, we present two perspectives on the system aimed at managing P^+ initiatives either via tracking current activities or via risk management.

SPI Tracking System (5p)

By taking a personalization perspective on S5 combined with a (in)formal way of tracking P^+ initiatives, the idea for the SPI tracking system arises.

The basic idea of the new system is to support improvement activities initiated by P^+ by means of well-defined SPI practices. In particular, keeping track of activities is of special interest.

The system is modelled as an interaction system because it suggests an improvement to the management of SPI activities. The CAIWOE and root definition of the SPI tracking system are the following:

CAIWOE: SPI tracking system (5p):

C: Members of P^+.

A: Members of P^+.

I: SPI activities managed through tracking ongoing efforts and detailed planning.

W: Systematized follow-up of ongoing SPI activities provides a more focused effort where initiated actions are not neglected or forgotten.

O: P^+.

E: The transformation must not require too much work overhead. The documentation of initiatives and the updating of progress must therefore not be too work demanding.

Root definition: SPI tracking system (5p):
A system to manage and follow-up on initiated SPI activities, by documenting, coordinating, evaluating and visualizing SPI activities, in order to systematize SPI efforts and to identify new problematic areas.

The conceptual model for the new system is shown in Figure 4.9.

SPI Risk Management System (5c)

By taking a codification perspective and combining this perspective with the idea of managing P^+ initiatives through SPI risk management, adopted from Iversen et al. [Iversen et al., 2002], a new system is proposed for managing risks involved in SPI. The codification strategy means that such risks are managed through codification of risk profiles encompassing risk items and resolution actions. As the work of P^+ mainly concerns experience sharing between project managers, P^+ may be viewed as some sort of SPI group, conducting several (informal) SPI initiatives. Such initiatives may be managed via SPI risk management.

The new system is modelled as an interaction system, managing risk items and resolution actions via SPI risk management. The new system is summarized in the following CAIWOE and root definition:

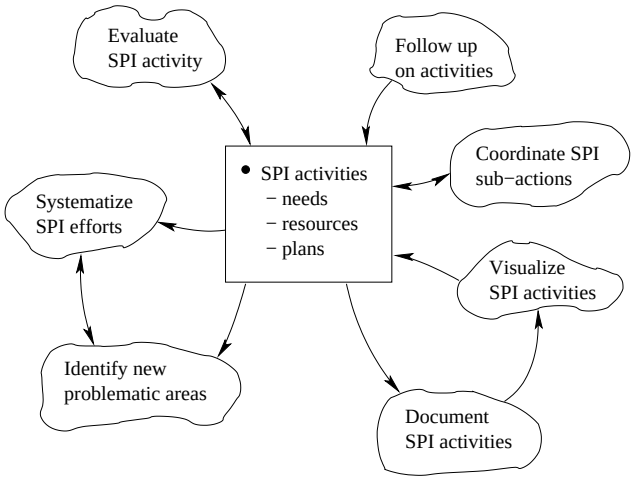


Figure 4.9: *Conceptual model of the SPI tracking system (5p).*

CAIWOE: SPI risk management system (5c):

C: Members of P^+.

A: Members of P^+.

I: Risk items and resolution actions managed through risk profiles of each activity.

W: A generic focus on risks involved in SPI activities is essential to the improvement effort and would be very beneficiary to both P^+ and project managers. In order to keep track of risks and focus on the importance of risk management during a predetermined period of time, risk profiles provide an overview of the risks characterizing the time period. Such profiles may be used to manage SPI activities.

O: P^+.

E: The use of risk profiles must not demand formalities that compromise the strengths of the informal nature of P^+. In this respect, benefits must follow immediately such that P^+ members stay motivated.

Root definition: SPI risk management system (5c):

A system to manage SPI risks involved in P^+ initiatives, by means of formulating and evaluating risk profiles, in order to identify problematic aspects of current practices and guide the SPI activities through resolving perceived risks.

The conceptual model of the new system is shown in Figure 4.10.

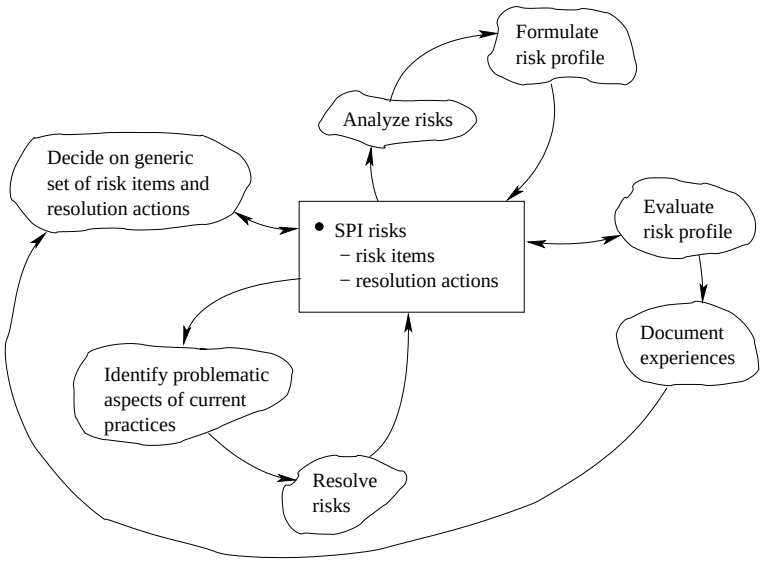


Figure 4.10: *Conceptual model of SPI risk management system (5c).*

4.4.4 Choosing Our Focus

We have now presented six new systems generated by applying various knowledge management perspectives to the three generic systems selected in the second iteration, see Section 4.3. In particular, personalization and codification were used as perspectives. Considering each of the six new systems in terms of their potential to improve or support P^+ initiatives, we limit our further analysis to include only four of the six systems presented in this third iteration of our SSM analysis:

- War stories system (1c)
- Agenda system (4c)
- SPI tracking system (5p)
- SPI risk management system (5c)

The reason for not continuing with the broker system (1p) is that it represents a formalization of the personal networks. The network must be combined in some way to form a network for e.g. the Systems Development branch. Thereby, the network would no longer be personal and the person dedicated to maintaining the network would not have a personal interest in such maintenance. Furthermore, it is unlikely that WM-data would set aside resources to incorporate such a change because it demands a lot of work from the knowledge broker.

The reason for not continuing with the reward system (4p) is that such kind of rewards are unheard of in WM-data. The introduction of rewards would generate heavy resistance among employees and conflict with the informal company culture which is based on people helping each other, *not* for personal gain, but rather as a shared appreciation of being part of something, e.g. a project group, and thereby feeling a responsibility for the success of the group.

Up until now, we have proposed systems assisted by knowledge management theory, analytical background knowledge gained through our previous study [Hosbond and Ørtoft, 2003] and through conversations with our contact person at WM-data. The next iteration of our SSM analysis presents the results of a comparison of our systems thinking and the real world, relating to stage five of the seven stage model. Obviously, in our case the real world represents the actual setting of P^+ in which the systems are to be implemented.

4.5 4th Iteration: Comparison

The fourth iteration of our SSM analysis presents the results of our comparison stage in SSM. Up until now, we have proposed and modelled relevant systems from an analytical point of view, believing that these systems could improve the initial problem situation, described in Section 4.2. Whether this belief is true or not has to be answered before initiating prototyping. We seek these answers through a discussion session with the problem owner, see Section 4.1.1, represented by our contact person whom we in this section, for sake of convenience, refer to as X. The main objective of the session is to compare the relevance and applicability of the four systems selected in our third iteration of SSM, see Section 4.4, with respect to actual practices in P^+.

4.5.1 Comparison Setting

Before presenting the results from the comparison session, we first sketch its setting. With respect to the participants in the comparison session, it was not possible to gather the entire group of P^+. Instead, the comparison was conducted with the participation of our contact person and ourselves. In this way, the comparison was performed cooperatively. One might think that involving only one person in the comparison limits its general applicability, but we argue that our knowledgeable contact person — a highly experienced project manager and a key member of P^+ — together with our background knowledge on P^+ and its political as well social environment, foster a satisfying comparison between the four selected systems of the third iteration and the real world situation of P^+.

Prior to the session, we defined a set of generic questions related to each of the four systems. These questions are described in the following subsection. Furthermore, we organized a set of worksheets representing our analysis of the four systems and

sent these beforehand to our contact person. By doing so, we aimed at ensuring his understanding of the systems along with their intention and meaning before initiating the comparison. In effect, this lessens the need for introducing and explaining each system, increasing the amount of time spent on comparing systems with the real world. The duration of the session was initially planned to take approximately 2 hours in order to ensure a qualitative discussion.

4.5.2 Model of Comparison

The comparison session is divided into four overall parts each related to one of the four selected systems. To guide the comparison of each system and the real world, we define a set of generic questions to be asked about each activity in the systems. After an evaluation of all the activities, we discuss the overall need for the system with respect to future application in P^+. The questions relate to how the activity is currently done, the feasibility of the activity and the obstacles that may stand in the way for adapting the new activities suggested by the system. We note that the questions are inspired by the questions in the comparison model presented by Checkland and Scholes [Checkland and Scholes, 1990].

The set of questions are split in two subsets. The first subset is used if the activity already exists and the second subset is asked if the activity does not exist. As noticed, both subsets focus on the feasibility of the activity.

The first subset relating to existing activities are formulated as follows:

- How is the activity currently conducted? (problems / success / resources)
- Is it conducted in a satisfying way? (why / why not)
- Any changes to the way the activity is currently conducted? (why / why not)
- Judge the feasibility of the activity? (why / why not)

The second subset includes questions relating to activities not currently present in P^+:

- Judge the feasibility of the activity? (why / why not)
- Suggestions for changes of the activity or an alternative activity? (why / why not)
- Any obstacles for the success of the activity? (why / why not)

The outcome of the comparison based on these questions are elaborated on next.

4.5.3 Comparison: Systems Thinking vs. Actual Practices

Based on the generic questions just presented, we initiated a comparison of the activities of the four systems and current practices of P^+. The following documentation consists of general conclusions and considerations with respect to the four systems, including quotations supporting these. We will not introduce a comparison of all the activities included in the systems; only those relevant to the overall evaluation of the systems are included.

As a general remark, we notice that the comparison did not result in any changes to the conceptual models.

Agenda System (1c)

On an overall level, the agenda system was well-acknowledged and the whole idea of letting P^+ members influence agendas of future meetings seemed interesting to X. The activities of the systems existed in some more or less structural form already. However, X stressed that one of the current problems related to the way agendas and meetings were conducted, concerned how the topics forming the next meeting were selected. The task of selecting topics for the next meeting was often in the hands of a particular P^+ member. Even though P^+ members had the possibility of suggesting topics by simply contacting the one responsible for the next meeting, this seldomly happened as emphasized by the following quotation:

> *"The selection process of agenda topics is not satisfying. The process needs to be changed."*

With respect to obstacles, X did not express any to the adoption and acceptance of the system as formulated in one of his sayings:

> *"I see no obstacles. If everyone proposes topics for the agenda, these topics should be prioritized. However, the task of prioritizing the topics could be assigned to another P^+ member every other meeting."*

War Stories System (4c)

The war story system seemed less intuitive to X and required further clarification before initiating the actual comparison. The term *war story* confused him, but after a short presentation where X tried to picture and present an example of a war story, X seemed to understand the basic idea. The activities of the war stories system were primarily non-existing in current practices. However, sharing of war stories (to X sharing of experiences and stories from practices) was currently an informal activity

that generally dominated the informal communication within P^+ and the project culture. This is emphasized in the following two sayings:

> *"Sharing of experiences and stories are currently done informally through P^+ meetings and sparring sessions. Project managers and P^+ members use these stories to describe practices from projects or experiences with the end-customer."*

> *"Sharing of stories is already an integrated part of our project culture and is what makes our project teams function very well."*

With respect to X's overall perception of the war story system, X expressed great interest and enthusiasm towards a future implementation of such a system. X mentioned that this was a way of capturing experiences which could be relevant to other project managers. But, he also emphasized that:

> *"War stories could lead to the identification of new problems.*

When discussing potential obstacles, X noted that the context of an informal story could be difficult to transfer and fully represent through codification:

> *"The task of representing the context in a codified war story seems difficult. Some specific structures must be explored."*

SPI Risk Management System (5c)

X's notion of the SPI risk management system was less successful. X understood the idea of the system and believed that the system could come in handy in the future when improvement initiatives like P^+ was properly accepted and anchored in the organization. However, X expressed concerns towards the level of current applicability. This was due to the administrative workload which he argued was required in order to benefit from the system. The extent of the workload simply called for resources that at the time would be impossible to provide. However, having said that, he also admitted that the way improvement initiatives were currently selected and initiated was unsatisfying. This is expressed in the following:

> *"It is not satisfactory that it is up to grassroot-driven initiatives like P^+ to come up with new initiatives that could improve existing practices."*

Furthermore, X argued that it would be in everyones interest if new initiatives were the result of an ongoing improvement effort conducted at a higher organizational level, thereby decreasing the problem of not being properly anchored in the organization. This is expressed through the following quotation:

> *"It would be nice if the formulation and selection of improvement initiatives was initiated by upper management or an institutional group like a SPI group."*

However, as relevant as the idea of the SPI risk management system might sound, X was still certain that the system would not fit the current work practices of P^+, but might instead stand the risk of compromising existing informal culture of P^+ by formalizing practices of P^+. To conclude his explanation, X also mentioned that focus and resources were at the time centered on conducting activities which had a visible and tangible outcome in the form of leverage, e.g. money. Therefore, resources are transferred to the projects and not to activities which WM-data cannot see the immediate benefits of, e.g. SPI-related initiatives such as P^+.

> *"It is difficult to introduce risk management in improvement initiatives with the same effect as in projects because it does not focus on an outcome in the form of money."*

SPI Tracking System (5p)

The SPI tracking system concluded our comparison session. X's perception of the system was very similar to that of the risk management system. X acknowledged the importance of a system to formally manage improvement efforts and believed it could come in handy sometime in the future when the process of managing improvement efforts had been properly anchored. But until then, X believed that the system would create a great amount of administrative overhead and put unnecessary constraints on the P^+ initiative, compromising the informal nature of the group. Again, the main obstacle for successfully implementing the system was related to the limited amount of resources and the non-existing anchoring of P^+ in the organization. This was expressed as follows:

> *"We [P^+] could easily use two days a month for managing our improvement efforts, but the required resources do simply not exist. Furthermore, both commitment and proper organizational anchoring of P^+ are absent."*

4.5.4 Results of the SSM Analysis

The result of our SSM process is twofold. First we have gained an understanding of the principles underlying SSM and we have experienced an actual application of SSM to a problematic situation from the real word. However, our process of SSM has been problematic with respect to our initial conception and use of SSM. This aspect will be elaborated on in further detail in the discussion, Chapter 6.

Second, considering the outcome of the comparison, we have found that the agenda system and the war stories system are highly relevant to the work of P^+. The agenda system can assist in creating a generic focus on the next P^+ meeting through a joint discussion on relevant issues, while the war stories system can support and improve the process of capturing, acquiring and sharing experiences. The two systems encompass analytical knowledge gained through the application of SSM, also taking into consideration the cultural aspects as analyzed in the cultural enquiry, Section 4.1.

Consequently, we initiate the next phase of our process of tailoring knowledge management systems by experimenting with the system ideas that can be formulated based on the two selected systems. That is, through a series of experiments based on evaluating prototypes, we intend to develop two prototypes based on the system ideas of the agenda system and the war story system.

Chapter 5

Prototyping

In this chapter, we describe the experiments conducted with the two selected systems from the previous chapter: The agenda system and the war stories system. We conduct two iterations. The first iteration is aimed at testing the system ideas that can be formulated based on the conceptual models. The second iteration aims at refining these system ideas by specifically addressing proposed modifications from the first iteration. We conclude the experiments by discussing the validity of the system ideas as evaluated during the experiments.

5.1 1st Iteration: Evaluating System Ideas

The first iteration of prototyping is guided by the main objective: Evaluating the system ideas of the two human activity systems described by the agenda system (Subsection 4.4.2) and the war stories system (Subsection 4.4.1). The correlation between the results of the SSM analysis and prototyping is thus the implementation of two prototypes to support these systems. In effect, we need to find out whether or not the systems provide useful support to the work of P^+. In this way, we aim to support activities of the conceptual models by implementing a prototype. However, the essential question is how to support the two systems.

The basic goal of our first iteration of prototyping is thus exploratory. Through experiments, we aim to explore two system ideas based on the two human activity systems and experiment with the kind of functionality required in order to tailor the prototypes to the specific needs of P^+. As part of the experiment, we aim to generate ideas for improvements and alterations that may improve the quality, relevance and applicability of the proposed prototypes. As a side-effect, we therefore want to determine what may be done to support the work of P^+ from a knowledge management perspective along with the types of requirements.

Due to limited resources in the form of contacts at WM-data and access time to test

persons, the prototyping of both system ideas has been performed as a single process. Furthermore, the experiments have only involved one test person. The reason is, that WM-data have only been able to participate in a limited number of experiment sessions due to workload of involved persons. Therefore, we have chosen to evaluate two prototypes during one session instead of making fever iterations. This fact influences the way the description of the first iteration is structured. Instead of describing each of the five stages of the process for each prototype, we describe the five stages jointly. The rationale for this choice is that many aspects of the two experiments coincide. Only when differing approaches exist between the two experiments, we describe them separately.

5.1.1 Planning

As stated above, the focus of both experiments is on evaluating the system ideas of the two human activity systems. These system ideas specify the basic ideas of the prototypes. Therefore, we first need to formulate the system idea in order to get a notion of what is being experimented with. The system ideas for the two prototypes are formulated as follows:

- ***Agenda prototype***: The system idea of the agenda prototype aims at facilitating the creation of shared agendas by providing access for group members to a web-based forum. Here, topics may be discussed, fostering commitment towards the meeting and establishing a shared context for participants.
- ***War stories prototype***: The system idea of the war stories prototype aims at supporting the codification of war stories. The prototype manages codified stories and supports the sharing of such stories by providing web-based forum in which the stories may be created. It is not the primary intention that the stories should be shared via the prototype, but rather, the phorum should act as a compository where inspiration may be sought.

Besides the main objective of experimenting with the above system ideas, the first iteration also includes a number of sub-objectives for each of the two experiments. These sub-objectives aim at answering the following questions during experimentation and evaluation of the prototypes:

- ***Agenda prototype***:
  - How may the use of an agenda contribute to the establishment of a shared context?
  - What should the agenda prototype provide in order to motivate P^+ members to affect the content of P^+ meetings?
  - To which extent may the agenda prototype increase the communication between P^+ members?

  - What kind of shared understanding does the agenda prototype help create about what is managed on P^+ meetings and how?
  - How will the agenda prototype foster an increased commitment for P^+ members towards P^+ initiatives?

- ***War stories prototype***:
  - What can be done to recreate the context in which the story was initially created?
  - How will the use of war stories affect the communication between P^+ members and project managers?
  - Do war stories succeed in helping to create a shared context between the teller and the listener?
  - Is it at all possible to codify war stories? and Why/why not?
  - How may the codification format be structured?
  - How may the enthusiasm with which the story is first told be captured in the structural format of the war story?

The above questions are helpful in guiding the experiments and provide a list of questions and concerns to be answered in subsequent evaluation.

Another aspect of the planning stage concerns the three terms *purpose*, *limitations* and *conditions* described in Subsection 2.4.3. These terms relate to the implementation of the prototype, not the whole experiment. It would be unnecessary, then, to document these aspects in detail and afterwards present the implementation. Therefore, we only document the implementation. However, we have noticed before developing the prototypes that the functionality required in order to make the *proof-of-concept* of both system ideas demands for both a very broad and in-depth implementation, meaning that most of the functionality suggested for by the conceptual models need to be implemented in order to make the evaluation of the system ideas less complicated.

5.1.2 Development

The development stage has been very rapid and took less than 80 man hours. Requirements for functionality have been derived from the conceptual models and the results of the comparison. In this way, the conceptual models have proven very useful.

System Architecture and Technology

Both prototypes have been constructed using the same technology. We use a client-server architecture based on HTTP. This means that the prototypes are web-based

and run via the Internet or intranet. The reason for this choice of architecture is that it enables us to easily fulfill two essential requirements of both prototypes: They must be distributed and easily accessible. Using web-based technologies thus facilitates both our development stage and the communication between users of the prototype, the first account satisfying the demand for rapidness. The programming language used for developing the two prototypes is ASP (*A*ctive *S*erver *P*ages) version 3.0. ASP is a small-scale programming language and is easily combined with the HTML framework. This enables us to develop our prototypes very rapidly which is the main reason for adopting ASP. Furthermore, WM-data's intranet services run on Microsoft Windows servers, including Internet Information Services, which makes ASP technology a reasonable choice. This type of technology has also influenced the choice of database technology as we use a Microsoft Access database for storing data. The database is accessed using the ADO component of ASP, making it portable to all Windows systems equipped with Internet Information Services. Because of the nature of the experiments, the prototypes are only run locally on a laptop. 4th generation tools such as Microsoft FrontPage and Microsoft Access have been used on occasion to support the development of user interface and the database, respectively.

With respect to the breath and depth of our implementation, the prototypes are a mix of both horizontal prototyping and vertical prototyping. This is required in order to make a *proof-of-concept*. We cannot limit the experiments to either a vertical approach or a horizontal approach because it is the system idea which we want to prove the feasibility and suitability of. Accordingly, much of the required functionality has been implemented. That being said, there are some parts of the prototypes that have not been fully implemented, e.g. the ability to send e-mails. For these parts we use stubs, i.e. pieces of code which emulate the requested functionality instead of providing it.

With respect to the design of the prototypes, both prototypes are based on the same graphical user interface and database component. This enables us to reuse components in both prototypes and would at a later point allow us to merge the two prototypes, should that prove to be of particular interest.

Design of Graphical User Interface

The design of the two prototypes is focused on making them as intuitive and user-friendly as possible. This means that the two prototypes have the same GUI and are similarly structured. The basic rationale is that the prototypes should be easy to use and not demand an unacceptable amount of overhead for the P^+ members. Figures 5.1 and 5.2 show examples of screen shots from the agenda prototype and the war stories prototype, respectively.

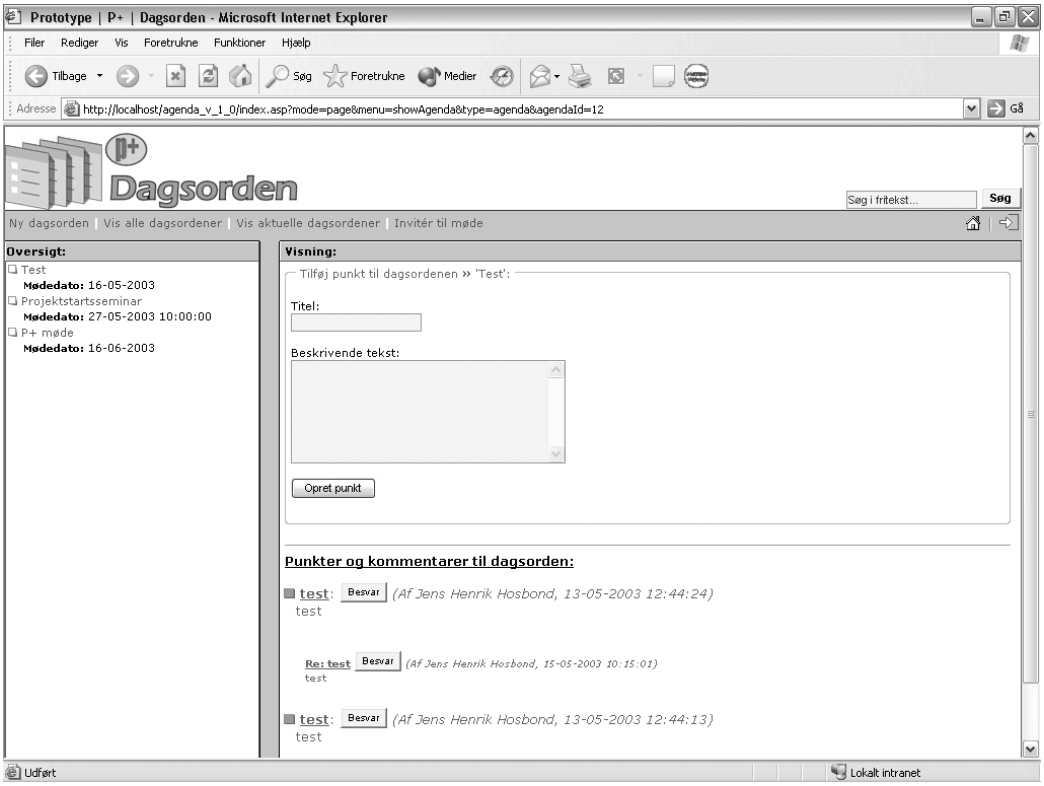


Figure 5.1: *A screen shot of the agenda prototype showing an example of the structure of a discussion.*

Functionality

The functionality and basic requirements for the two prototypes has been derived from the conceptual models. Instead of providing a full list of implemented functionality, we illustrate parts of the functionality via examples:

- ***Agenda prototype*:** As can be seen from the screen shot of Figure 5.1, basic functionality for managing agendas has been implemented. This includes the ability to create an agenda, show agendas and invite group members to a meeting (stub). Furthermore, Figure 5.1 shows the possibility of proposing an item for the agenda. Such items may be commented on by answering an item. Another feature concerns the ability to search agendas for certain words of phrases. In addition, the prototype includes functionality for managing users and keeping track of user contributions.

- ***War stories prototype*:** As seen from Figure 5.2, basic functionality of the war stories prototype includes the creation of war stories, creation of overall

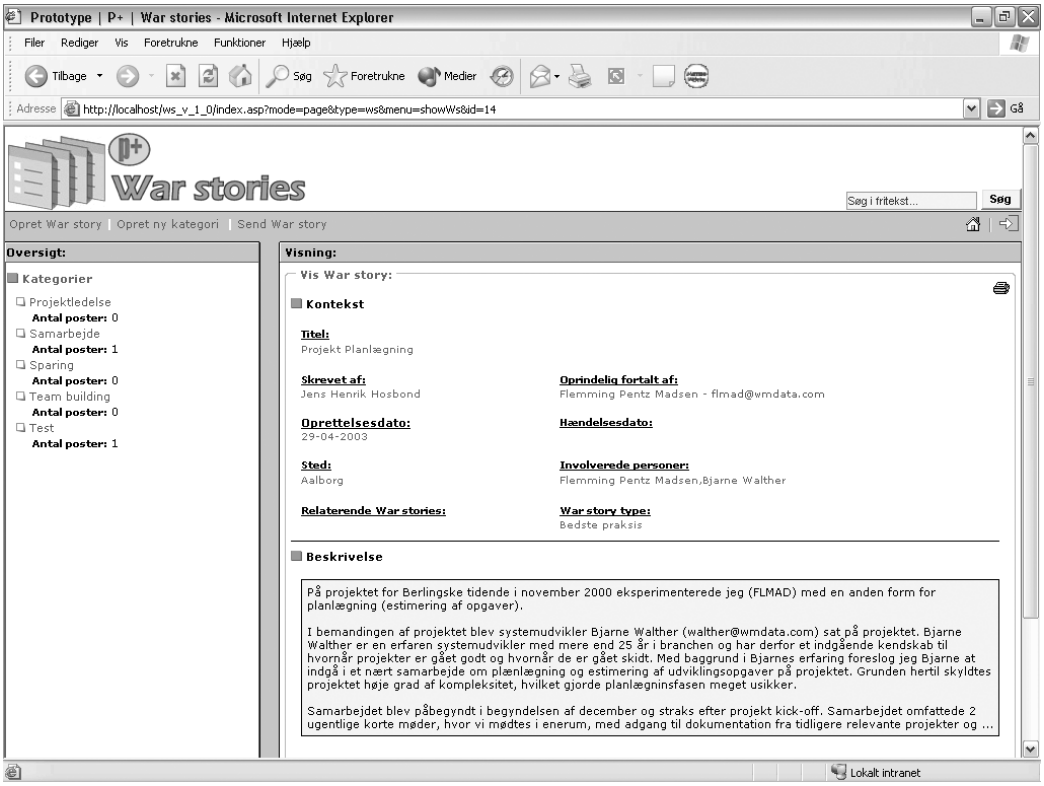


Figure 5.2: *A screen shot of the war stories prototype showing the codification of a war story.*

categories and the ability to send a war story (stub). Furthermore, it is possible to perform a basic search through the war stories for the occurrence of a specific word or phrase. As with the agenda prototype, the war stories prototype includes basic user management functionality for tracking user contributions.

Codification of War Stories

One problematic aspect of capturing war stories in a computer application is the codification. How may such war stories, which often includes a highly specific context, be stored in an application? and How may such context be codified? In this first iteration, we try out a rather simple codification of war stories by allowing the war story to be described in full length. Besides the description, we extract certain attributes of the war story, including title, author, original story teller, date of documentation, date of event, place of event, involved persons, related war stories and type of war story (best, worst, other). The last account relates to a simple categorization of the war story as either an example of best practice, worst practice or other. As stated

in the description of the planning stage earlier in this section, Subsection 5.1.1, one of the sub-objectives of the war stories experiment is to generate ideas for a suitable structural format for the codification of war stories. This result is elaborated on in Subsection 5.1.4.

5.1.3 Preparation

In order to set up the experiment, we need to define our cooperation with the test person, the degree of realism during the experiment session and the selection of users as described in Subsection 2.4.3. These aspects of the experimentation are addressed in the following paragraphs.

Cooperation

Our cooperation with the test person takes the form of a very loosely structured discussion during which answers to the questions stated in Subsection 5.1.1 are sought. The discussion is open-ended only guided by the test person's exploration of the prototype and the direction set out by the general questions. During the experimentation, the test person must be given the freedom to explore the prototype on his own. Furthermore, he is encouraged to 'think out loud' and comment on what he experiences. Problems and suggestions for modifications and alterations are discussed when they occur along with the usefulness of existing functionality.

Realism

The prototypes are tested in a detached environment, away from the test person's normal work environment. However, as both future application may be used anywhere with access to the Internet, the experiment setting is somewhat realistic. Another possibility would be to make the test in different environments according to future use situations. However, this is not important to the *proof-of-concept* because our test person is able to abstract from such details due to his experience in systems development and use of computer applications. A realistic situation from daily practices is not set up and no use cases or user stories are constructed to guide the experiment. In this way, the experiment relies on the explorative capabilities and interests of the test person. We find this approach reasonable as our test person is an experienced systems developers accustomed to testing and evaluating applications. Consequently, we argue that he is able to abstract from the prototypes and evaluate the underlying system ideas. We refrain from further discussion of this argument and elaborate on the issue during the presentation of our results on user involvement in our experiments, Section 6.2.2.

Selection

The test person is our cooperative contact at WM-data whom we have had several prior meetings with during our research period. He is a key member of P^+ due to a high degree of personal interest and has been involved in the P^+ initiative from the beginning. Therefore, he is capable of judging whether or not the support provided by our prototypes are of any use. Furthermore, the test person has been involved in the comparison stage of our SSM analysis described in Subsection 4.5.3. With respect to the experiment, he does not have to prepare anything in advance, but rather show up with an open mind.

5.1.4 Experimentation

The experimentation took place as a cooperative evaluation of the two prototypes using an open-ended approach. The prototypes were experimented with sequentially so that only one prototype was evaluated at a time. First, the test person was given a short demonstration of the prototype. Afterwards, he had the opportunity to explore the prototype on his own while we sat besides him. We only interfered when he seemed stuck which only happened once. The experimentation and the subsequent evaluation were documented by taking notes. The use of a tape recorder was decided *in situ* not to be necessary. After the experimentation, we conducted a cooperative evaluation of each prototype in the form of a debriefing where we ensured that the questions comprising the sub-objectives were answered, see Subsection 5.1.1.

The results of the experimentation and the evaluation have been documented in a list of comments and key statements. Based on this list, we address the comments made with respect to the system ideas, including important aspects of the sub-objectives relevant to the system ideas. In each paragraph, this overall evaluation of the system idea is followed by a list of specific comments regarding details of the prototype along with justification for each comment. We first turn to the agenda prototype.

Agenda Prototype

On an overall level, the system idea represented by the agenda prototype was considered to be highly relevant to the work of P^+. The test person remarked:

> "*Organization of agendas is clearly in demand because someone has to take care of making an agenda. Meetings without agendas are unacceptable because they start off without focus. Therefore, the agenda system represents a good supplement to the planning of meetings.*"

The discussion of proposals for agenda items was viewed as very useful because it could ensure a shared understanding of items prior to meetings. When the final

agenda was to be created, the comments to each item could then help judge the importance of the specific item and possibly result in changes. Furthermore, the test person saw the establishment of a shared context prior to the meetings as very beneficiary, as the following quotation shows:

> "*By using the agenda system, people would be more focused when meetings start. Furthermore, the participants would have a shared understanding of what the topics are for the meeting and what is going to be addressed.*"

Furthermore, in the opinion of the test person, the system idea would also prove useful to experience groups which also need a way to focus their meetings. This is currently problematic because meetings are rare and their members come from different geographical locations.

With respect to the design of the graphical user interface, this was found satisfactory. The test person appreciated the design and noticed that:

> "*it was pleasing to look at and very easy to grasp.*"

With respect to details of the implementation, the test person noticed the following accounts:

- It should be possible to piece together the agenda based on the underlying proposed items of the discussion.
- After piecing together the agenda, it should be possible to prioritize items so that less important topics occur last. This is relevant because not all topics are covered during P^+ meetings and the less important ones are either disregarded or moved to the next meeting.
- The future application should be adopted to fit the intranet platform at WM-data, namely Microsoft Exchange Server and Microsoft Outlook. Thereby, the P^+ members would not think of the future version of the agenda prototype as "*yet another application*".
- Another desirable functionality would be to be able to attach documents to agenda items and comments. This is a feature often used in their current work and would provide greater means of expression in the agenda prototype.

The resulting actions from this experiment are described during the evaluation of the experiment, Subsection 5.1.5.

War Stories Prototype

The underlying system idea of the war stories prototype was also judged highly relevant. Among other things, the test person remarked the following:

> "*The war stories reflect a common set of experiences and are a concrete expressions of the company culture. They are an inherent part of our project culture. Experienced co-workers thus utilize war stories to a great extent in their work.*"

Furthermore, the codification of war stories via the war stories prototype was seen as a very positive attempt at sharing such stories. The tool was judged easy to use and the administrative overhead introduced in documenting stories was acceptable.

Another remark related to the personal networks:

> "*Overall, the prototype would provide great support for the personal networks. Especially less experienced project managers could use the prototype to find key persons via the prototype that possess particular knowledge, and then use the prototype to locate e-mail addresses and thereby establish contact.*"

In this way, the war stories system could function as an *indirect* competence system because authors and storytellers are represented in the story as attributes of the context. In this way, the people who experienced the events of the story may be sought. Furthermore, it was noticed by the test person that:

> "*it would be useful if the story contained an explicit reference to the project where it took place.*"

The explanation given is that the impact of the story is established through such a reference, although such relevance decreases across time. Another improvement concerned the possibility to aggregate stories and thereby package different topics into a single war story. As with the agenda prototype, the war stories prototype could be used outside of the work of P^+. For instance, it could be used in correlation with "Ratten" and kick-out meetings.

With respect to details of the implementation, the test person noticed the following accounts:

- First of all, the test person remarked that it would be relevant to be able to distinguish between primary and secondary category of a war story; most war stories would have relevance in several categories so the prototype should be able to handle more than one categorization for each story.

- Furthermore, the structural codification format of the war stories was commented on:
  - The format could include an explicit point or morale, e.g. in the form of a 'lessons learned'. This would provide a useful extension to the current format according to the test person.
  - Stories could be partitioned into explicit parts: What, why and how. This would make for a clear division of relevant accounts of the story.
  - Possibly, stories could follow an explicit template. However, this template should be easy to follow and very intuitive.
  - The description of the story should include a set of keywords summarizing the content of the story. As a side remark, it was mentioned that these keywords should be available as criteria when searching for war stories.
- The overview of the content of each category should show the title and lessons learned of each story, and not just the title.
- A more advanced search functionality was required, allowing the various attributes of the war story (author, persons, keywords, lessons learned etc.) to be searched.

5.1.5 Evaluation

The results of the debriefing as described in the previous subsection have subsequently been evaluated. Based on this evaluation, we have decided on what to improve for the second iteration. We refrain from further description of these changes because they are described in detail in the documentation of the second iteration, Section 5.2. With respect to the system ideas, our test person found them highly relevant to the work of P^+. Instead of addressing their validity here, we discuss the most interesting aspects in Section 6.3. This discussion includes relevant aspects of the sub-objectives stated in Subsection 5.1.1. These sub-objectives were not addressed directly during experimentation, but rather, they served the purpose of assuring the validity of the underlying system ideas of the prototypes. We now turn to the second iteration of prototyping.

5.2 2nd Iteration: Refining System Ideas

Having focused on the system ideas of the two prototypes during the first iteration of prototyping, the second iteration aims at refining the system ideas by implementing the proposed modifications. Thus, the main objective is to evaluate the benefits provided by these modifications and determine whether or not they match expectations. This objective influences the goals of the iteration. Whereas the goal of the first iteration has been exploratory, the goal of the second iteration is rather a combination

of exploration and evaluation. Still, we want to explore other options, but we also want to evaluate the changes made to each prototype.

Besides the main objective, ulterior sub-objectives also exist as presented in the descriptions of the planning stage. These sub-objectives address details of the implementation. Again, the two prototypes are evaluated during one experimentation session, influencing the structure of this section. Each stage is described and when there are differing remarks to be made on the two prototypes, these remarks are clearly separated.

5.2.1 Planning

The sub-objectives for this iteration aim at answering the following questions:

- ***Agenda prototype***:
  - To what extent may ideas and thoughts on topics be shared in order to create a shared context for participants?
  - Is there a limit to the level of detail of comments?
- ***War stories prototype***:
  - Is there other ways to design the structural format?
  - How is the partitioning into 'what', 'how' and 'why' perceived?
  - To what extent is it possible to codify war stories?

5.2.2 Development

The development stage for the second iteration has also been very rapid and took less than 32 man hours. The improvements suggested during the experimentation in the first iteration, see Subsection 5.1.4, have resulted in the implementation of additional functionality, described in the following.

Functionality

In addition to the functionality described in Subsection 5.1.2, the following functionality has been implemented for the second experiment:

- ***Agenda prototype***:
  - piece together agenda before sending out the invitation, including functionality for prioritizing agenda items, see Figure 5.3
  - attach documents

  - edit agenda

- ***War stories prototype***:
  - view of search results must include the related category
  - view of categorized war stories in overview should include title + lessons learned
  - additional search attributes of war story (keywords, lessons learned, project reference, category and authors)

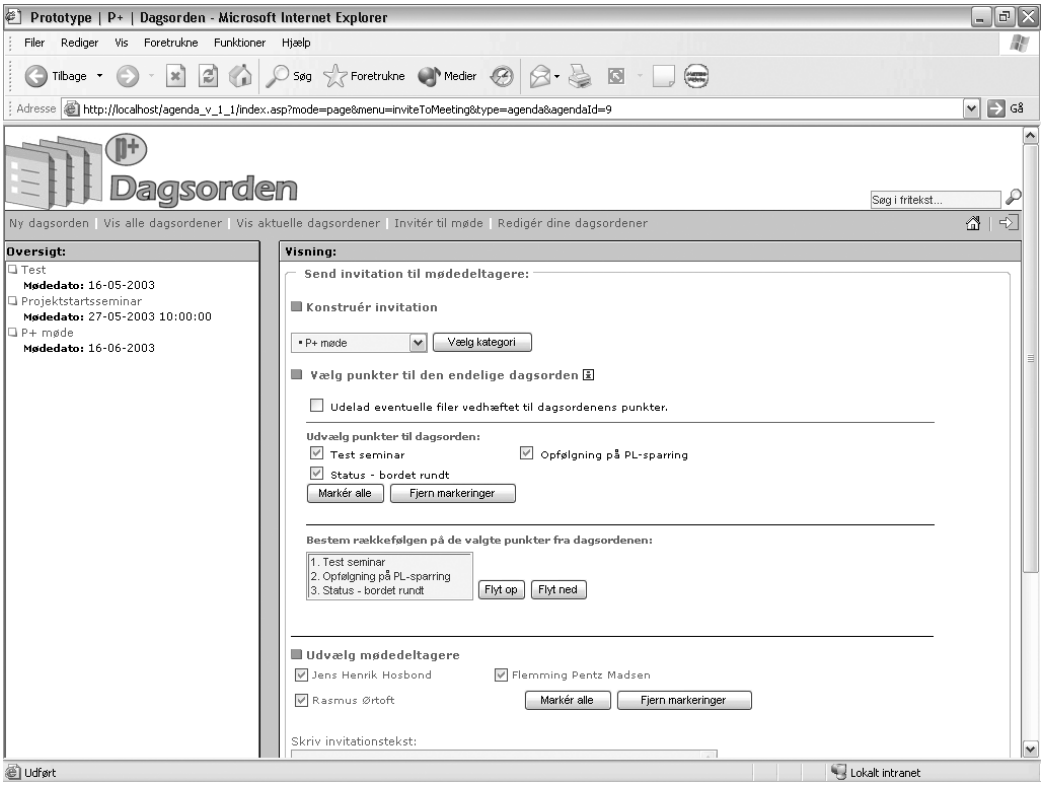


Figure 5.3: *A screen shot showing the prioritization of items for the final agenda.*

In addition to the above functionality, we have also made modifications to the codification format of the war stories.

Codification of War Stories

The codification of war stories was first altered such that war stories should be split into three parts: 'What', 'how' and 'why'. However, early tests indicated that it was rather hard to divide a war story into these parts; a place to actually tell the story

was needed. Therefore, we have kept the possibility of writing the entire story and made the fields of 'what', 'how' and 'why' optional. In this way, the epic effect of the war story as a coherent story is not compromised, but at the same time, the possibility remains for filling out the fields of 'what', 'how' and 'why'. The format is shown in Figure 5.4. Furthermore, a field for the lessons learned has been added, making it possible to summarize what has actually been learned from the incident described by the war story. Furthermore, a field for stating descriptive keywords has been added along with a field for adding the project reference.

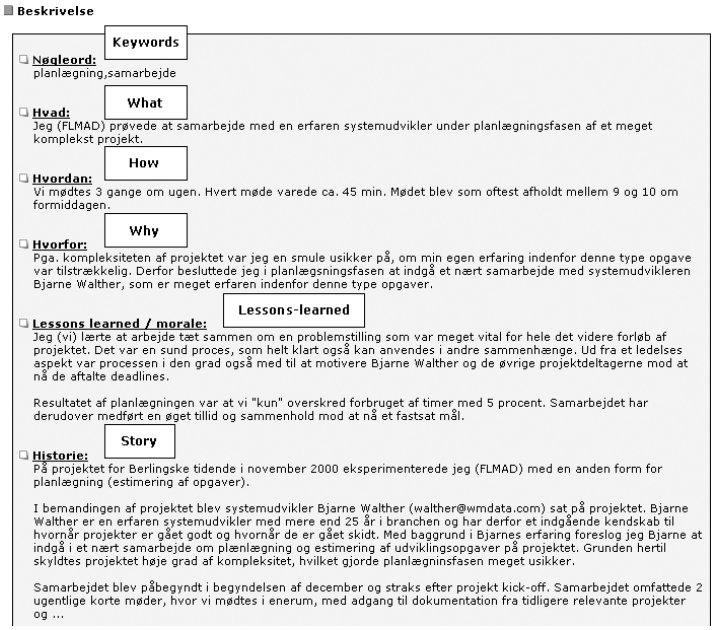


Figure 5.4: *A screen shot showing an example of the codification format for a war story. The screen shot does not include the attributes situated above the description. The text in the white boxes is the English translation of the Danish terms.*

5.2.3 Preparation

The preparation prior to the experimentation stage is similar to the preparation made in the first iteration, see Subsection 5.1.3. We therefore refrain from further description of the preparation.

5.2.4 Experimentation

As in the first iteration, the experimentation took place as a cooperative evaluation based on an open-ended approach. The prototypes were experimented with in sequence after a short presentation of the new set of functionality. The test person

was then free to experiment on his own while commenting his experiences. Again, no tape recorder was used because the experimentation took place at a pace that allowed us to take detailed notes. The results of the experimentation and subsequent debriefing are described in the following paragraphs.

Agenda Prototype

With respect to the overall design and the test person's overall impression of the prototype, he remarked the following:

> "*I think that we can easily use this. It could actually be very interesting.*"

Actually, he was rather enthusiastic about the prototype. However, he noticed the difficulty in trying out the system idea in a realistic setting, i.e. P^+ members using the prototype prior to a P^+ meeting. One major challenge would be to convince the other P^+ members of the usefulness of the prototype because they have not yet experienced the difficulties of making agendas (this has primarily been done by the test person). He, however, would like to see the prototype in use because he feels that it would improve the quality of the agendas and make the meetings more focused. Furthermore, he thinks that meetings would start off with more enthusiasm if everyone has a shared understanding of what is going to be addressed during the meeting; maybe participants have already thought about the items on the agenda and started their thinking process. In this way, the participants would be more focused on the issues of the meeting. However, this point needs to be further experimented with. Because of the limited time frame of this study, we have not been able to include such experimentation in our prototyping. The test person remarked the following:

> "*One of the obstacles for the adoption of the agenda prototype is that people must have experienced the problem of generating group agendas. They must be convinced that the application meets their demands and needs. People must be able to see the necessity of repeatedly tending to the preparation of agendas. Therefore, it is problematic to decide on the usefulness of the agenda prototype, as with all other groupware applications.*"

Another aspect of the experimentation concerned the implemented functionality. It was noticed by the test person that it was still highly relevant to adopt the agenda prototype into their existing Microsoft Outlook environment. In this way, the P^+ group requires additional calendar functionality of the agenda prototype. Specifically, they need the prototype to book meetings in their respective calendars. If one member is unavailable, the meetings must be rescheduled.

Other functionality concerned the possibility of handling more that one group. This is not currently part of the prototype, but could easily be handled by modifying our

login module. As a passing remark, the test person noticed that the prototype should be capable of assigning one person to more than one group.

Another aspect is the specific need to generate previews of the final agenda before sending it to the other P^+ members. Such a feature were advertised for by the test person and would be highly relevant. Furthermore, he suggested that the final agenda could be generated in a format other than HTML, e.g. Adobe's pdf. The support of the pdf format would, according to the test person, increase the usefulness considerably.

War Stories Prototype

The overall design of the war stories prototype was also to the test person's liking. The problematic aspects of the agenda prototype — the fact that it needed to be experimented with as groupware — did not seem as relevant with respect to the war stories prototype because, as the test person noticed:

> "*The war stories highly resemble what we are doing in daily practices. The goal must be to communicate a structural format of the stories that is adapted to the organization as a whole.*"

In this way, the test person suggests that the codification of war stories must suit the needs and demands of the company. However, the precise structural format of the war stories would have benefited from a more thorough investigation in order to be fully accepted by all P^+ members as the best way to codify war stories. Here, we cannot rely much on literature because very little is published on this particular aspect of storytelling.

Detailed comments with respect to the codification format regarded the use of 'what', 'how' and 'why'. First, the correct semantics must be established because the terms may be interpreted differently. For instance, does 'how' relate to how the incident described by the story happened or does it relate to how the incident was solved. Such ambiguity can be very misleading when stories must be codified. Currently, such ambiguities are addressed via popups where the semantics of the terms is explained.

Second, one must be careful not to repeat aspects of the story in the three fields. "*This is actually not an easy task*" as remarked by the test person. He further noticed:

> "*The use of 'what', 'how' and 'why' must be tried out in practice. If people write non-comparable descriptions in these fields, either the semantics of the fields may be more clearly defined or they can be completely removed.*"

Third, the size of the 'what', 'how' and 'why' fields should be made smaller, indicating that only a short description of these aspects is called for. With respect to the 'lessons

learned' field of the war story, the emphasis on the morale of the story was seen as very valuable. Hereby, one would quickly be able to extract the point of the story which is particularly useful when browsing for relevant stories.

More precise comments regarding the functionality of the war stories prototype first of all concerned the ability to delete war stories and not just be able to edit them. This facility should be limited to the author and enabled during editing. Additional functionality could include backup of deleted war stories, stored away under an extra menu item, 'Old War Stories'. Also, it should be possible for the user to restore such old war stories if they become relevant again. Another detail concerned the ability of altering the attributes 'original source' and 'project reference'. It should not be possible to alter these two attributes.

With respect to the search functionality, the search in the war story description was erroneous. In addition, the advanced search should include the facility of stating several keywords at once with the possibility of searching for either of these. In addition, one should be able to choose between different boolean semantics in the advanced search, e.g. AND and OR, when constructing the search.

Minor details concerned the following:

- The popup showing the short version of the war story should be bigger so that more of the war story is shown.
- The popup information on 'keywords' was missing.
- In the showing of the war story, the description should be moved in front of the 'what', 'how' and 'why' fields because "*it is the description which is important in the search for something interesting and it indicates a greater importance related to the three fields*" as remarked by the test person.

As with the agenda prototype, the test person requested the possibility of generating pdf versions of war stories. Other formats discussed included Microsoft's PowerPoint and Word formats. In this way, war stories could more easily be shared on seminars and meetings without accessing the prototype. As the test person noticed:

> "*The extraction of knowledge is important to the individual, but the knowledge extracted must be shared with others so therefore the presentation is important.*"

As a last request, we discussed the possibility of providing well-written war story examples. Such stories could serve as guidelines, exemplifying good war stories. A link could be provided from the dialog window where the war story is created.

5.2.5 Evaluation

After having analyzed the comments made during the experimentation and debriefing, we have decided to end the experiment. Time does not allow us to address further specifics of the prototypes. Furthermore, both system ideas expressed through the prototypes are found satisfactory by the test person. However, an important aspect to elaborate on would be to experiment with the prototypes by involving other P^+ members, evaluating the prototypes as groupware. Sadly, such an experiment could not be set up within the time frame of our research period.

With respect to the sub-objectives stated in Subsection 5.2.1, these have, as in the first iteration, been used to validate underlying aspects of the system ideas. This evaluation is addressed in Subsection 6.3.1. In this way, the system ideas are strengthened. Based on our prototyping phase, we therefore evaluate the experiments to be successful, having showed the relevance of the prototypes. With respect to the overall evaluation of the system ideas, this is addressed separately in Section 6.3. We now turn to a discussion of our results.

Chapter 6

Results

In this chapter, we present the results of our research. Our research approach has been aimed at answering the research question stated in the introduction, Chapter 1:

> **Research question:**
> *How can the process of tailoring knowledge management systems benefit from using Soft Systems Methodology and prototyping?*

In particular, we address the following areas:

- Acquisition of theoretical and analytical background knowledge, Section 6.1
- Methodological combination of SSM and prototyping, Section 6.2
  - Application of SSM
  - Application of prototyping
  - Combination of SSM and Prototyping
- Usefulness of our system ideas, Section 6.3
- Combination of hard and soft systems thinking, Section 6.4

6.1 Background Knowledge

The background knowledge established prior to this study showed to be very relevant with respect to our understanding of knowledge management theory and on the difficulties of applying relevant theories in practice. Furthermore, the findings of our case study from the previous research period provided a basis for initiating the SSM analysis presented in Chapter 4. These two aspects of our background knowledge are addressed in the following.

6.1.1 Knowledge Management Theory

First, we found that a solid epistemological understanding of knowledge was useful. This has provided us with an appreciation for various definitions of knowledge and the relationship between knowledge, information and data. Such basic knowledge was important in grasping the concept of codification which especially showed to be relevant in our development of prototypes for supporting knowledge-related activities.

Second, theories on various types of knowledge were relevant when trying to understand the different knowledge demands related to systems development in WM-data. We found Nonaka's distinction between tacit and explicit knowledge important, but also Mathiassen et al.'s four knowledge types in systems development: Situated, exemplary, community and procedural knowledge were found useful in understanding the kind of knowledge that are predominantly used in WM-data [Nonaka, 1994] and [Mathiassen et al., 2003]. In effect, this provided us with a notion of the type of knowledge that our prototypes were to support. In this case, we found that WM-data predominantly used tacit knowledge which is created through spontaneous situations of informal communication, thereby creating situated knowledge in the words of Mathiassen et al. [Mathiassen et al., 2003]

Third, the theories concerning knowledge conversion proved useful in understanding how current practices in WM-data convert knowledge, e.g. tacit to explicit or vice versa. Furthermore, the theories lead to an interest in whether or not these theories were at all possible to support via a knowledge management system. Nonaka provides the SECI model which represents a well-founded theory concerning the dynamics of organizational knowledge creation [Nonaka, 1994]. In addition to the SECI model Nonaka and Konno, have proposed the concept of *ba* which represents a shared space in which knowledge can be converted, Section 3.1. We encountered problems with expressing the conversion modes through the functionality of the prototype. Instead, we focused on establishing *ba* such that the prototype became the shared space in which knowledge was shared, e.g. the agenda prototype provides functionality for initiating debates which may create new knowledge.

Fourth, theories concerning communities-of-practices, Subsection 3.1.3 was found useful in the SSM analysis, Chapter 4, where they provided a theoretical perspective on the systems development practices at WM-data. This added to the general understanding of WM-data as an organization utilized especially during the cultural enquiry, Subsection 2.3.4. Here, the understanding of roles, norms and values was facilitated through the characterization of communities-of-practices from Brown and Duguid's three perspectives of *working*, *learning* and *innovation* [Brown and Duguid, 1991]. In particular, the use of narration was coupled to WM-data's use of storytelling as an important means of communication. This, in effect, formed the basis for the war stories system, Subsection 4.4.1, which showed to be highly relevant for P^+ during the comparison, Subsection 4.5.3.

Finally, a general appreciation of the fact that knowledge management does not exist in a vacuum enabled us to think of knowledge management systems in a much more broad sense than just applications. Hansen et al.'s distinction between the codification strategy and the personalization strategy provided relevant perspectives on the three generic systems selected for further analysis as a result of the second iteration (S1, S4 and S5), Section 4.3. In particular, codification and personalization resulted in the modelling of two human activity systems for each generic system, thereby providing differing perspectives on the generic system in question, see Section 4.4. In addition, Zack's work on developing a knowledge strategy affected how we perceived the strategic element of knowledge management. The importance of adopting the knowledge strategy to the company culture and the overall business strategy provided guidelines which we could follow when modelling our systems. Specifically, this concerned the tailoring of systems to the informal company culture of P^+ in our third SSM iteration, see Section 4.4.

Many other theories on knowledge management exist, but the areas discussed above served our purposes well. In addition, we found that knowledge on specific implementations of knowledge management systems provided different perspectives on the opportunities and threats in knowledge management. We benefited greatly from studying various reports on prior implementations and learned from their successes and mistakes. For instance, knowledge on specific knowledge management implementations was utilized in the modelling of the broker system, Subsection 4.4.1, which originated from an idea adapted from Hellström et al. [Hellström et al., 2001].

6.1.2 Applying Findings from Case Study

The analytical knowledge gained through the qualitative interviews served as a rich footing which was very useful to us during the present study. Combined with the theoretical knowledge, it provided us with different perspectives on knowledge management that we drew upon particularly in our SSM analysis, Chapter 4. This analytical knowledge also came in handy during prototyping where statements from the interviewees helped shaping particular aspects of the two prototypes. Specifically, general comments as to the applicability of the future applications guided part of our design process. This is illustrated in Subsection 5.1.2 where it is argued that the systems must be easy to use and not demand too much extra work. In the sections concerning the discussion on SSM and prototyping, subsections 6.2.1 and 6.2.2, respectively, we discuss in further detail how these phases of our research benefited from the knowledge gained through our previous study.

6.1.3 Consequences

As a consequence, the combination of both theoretical and analytical knowledge prior to the actual process of tailoring knowledge management systems proved essential

to our understanding of knowledge management-related processes at WM-data. It provided us with a rich understanding of both the company culture and the systems development practices at WM-data that specifically formed a basis for our cultural enquiry. Furthermore, it helped to focus our SSM analysis in the respect that we started off with a much more solid understanding of the practices at WM-data than is usually the case when conducting an SSM analysis. This also lead to some problems with adopting SSM that are discussed in Subsection 6.2.1.

6.2 Combining SSM and Prototyping

The combination of SSM and prototyping in the process of tailoring knowledge management systems was inspired by Mathiassen and Stage's 'Principle of Limited Reduction' [Mathiassen and Stage, 1992] as described in the introduction, Chapter 1. However, the combination of SSM's analytical approach and the experimental approach of prototyping brought up several interesting issues which are discussed in the present section. In particular, the issue of how the results of the SSM analysis are utilized in prototyping is of special interest. Before presenting our results in this area, we turn to our separate experiences of applying SSM and prototyping.

6.2.1 Application of SSM

In the following paragraphs, we present our experiences with respect to our use of Checkland and Scholes' two streams of enquiry, namely the cultural enquiry and the logic-based enquiry [Checkland and Scholes, 1990]. With respect to the cultural enquiry, we address the following two issues:

- Adding Analysis Four to the cultural enquiry
- Utilizing qualitative interviews

With respect to the logic-based enquiry, we address the area of:

- Applying interaction systems

Using Cultural Enquiry

When tailoring knowledge management systems, we found it of great importance to first understand the characteristics of the organization, e.g. the company culture, communication style, knowledge activities, knowledge levels and knowledge needs, see Section 4.1. Such findings gave a notion as to the kind of knowledge strategy existing in the organization. An understanding of the underlying knowledge

strategy, e.g. codification strategy or personalization strategy proposed by Hansen [Hansen et al., 1999], was useful in order to design and implement knowledge management systems tailored to the characteristics of the organization. It is exactly in gaining this understanding that the cultural enquiry, adopted from Checkland and Scholes [Checkland and Scholes, 1990], showed to be useful, namely by structuring our understanding of P^+. In addition, we came to understand that our cultural analysis was initiated much earlier than that of our actual SSM analysis, namely indirectly as part of our previous study [Hosbond and Ørtoft, 2003].

Adding Analysis Four to the cultural enquiry
Working with the cultural enquiry inspired us to add a fourth dimension, see Subsection 4.1.4. We found that the current distinction between the three analyses (roles, the social system and the political system) was simply not enough when analyzing problematic situations within knowledge management. The cultural enquiry also had to reflect how knowledge is managed and the characteristics of the existing knowledge management strategy, whether formal or informal. We therefore added a fourth dimension, Analysis Four, titled *knowledge* which described knowledge aspects of the organization, e.g. how knowledge is shared and created.

Utilizing qualitative interviews
Through the empirical investigation that formed our previous study, we conducted qualitative interviews with three institutional roles: Systems developer, project manager and senior manager. We found that the use of qualitative interviews resulted in an increased understanding of not just P^+, but also the environment surrounding P^+. Consequently, this lead to new perspectives on systems fitting the characteristics of the environment, but at the same time capable of improving the problematic situation. Either way, the conduction of qualitative interviews with various roles only enriched the understanding of the problematic situation which the systems were to support or improve. A thorough understanding of the problematic situation and its surrounding environment thus positively affected the outcome of the SSM analysis.

Using Logic-based Enquiry

The understanding of P^+ and its surroundings lead to the identification of problematic situations. This initiated the logic-based analysis following the seven stage model described in Subsection 2.3.3. The logic-based enquiry enabled us to view the problematic situation from different perspectives, leading to different systems capable of improving the situation. The various systems ensured that we did not limit our analysis.

Even though the stages of the seven stage model corresponded to our needs, we encountered difficulties in defining the transformation for some our systems. We therefore modelled four of the six systems as interaction systems, following Mathiassen and Nielsen [Mathiassen and Nielsen, 1998].

Applying interaction systems
We attribute our difficulties with defining transformation systems, which modelled human activities, to the area that we were trying to improve: Practices within P^+ related to knowledge management. How can a transformation with respect to an intangible commodity like knowledge be defined? What is the initial state? and What is the outcome? Due to our troubles, we found that expressing systems for managing knowledge by means of transformation systems was difficult due to the missing identification of inputs and outputs. We therefore found it useful to apply interaction systems in four cases, namely in the broker system (1p), the reward system (4p), SPI tracking system (5p) and the SPI risk management system (5c). The systems are described in Section 4.4.

The Usefulness of SSM

Initially, we experienced difficulties with applying SSM. Specifically, we found it confusing to consider the problematic situation from various angles instead of decomposing the problem as in traditional hard systems thinking. However, by working with the ideas of SSM and applying them on our already existing background knowledge of WM-data and P^+, we experienced an enriching learning process in which we came to understand and apply soft systems thinking. Even though we experienced an increasing understanding of soft systems thinking, it was a slow learning process and we believe that without the background knowledge on WM-data, the process would have been even slower. However, at the same time, we believe that our background knowledge possibly acted as an obstacle and affected our SSM analysis in the way that our background knowledge limited the degree to which we could maintain an open mind as required by SSM in order to treat all systems equally important.

In addition, we found that initiating an SSM analysis on an organization, e.g. WM-data, within the time span of four months, as in this study, requires either a thorough understanding on how to apply SSM or at least a thorough understanding of the characteristics representing the organization.

6.2.2 Application of Prototyping

When viewing the prototyping phase of our project separately, our adaptation resulted in numerous experiences. The following four areas concern experiences which we have made during our adoption:

- Organizing a rigorous process
- Utilizing background knowledge
- Involving test persons

- Establishing a shared context

In the following, we address three of the above areas.

Organizing a Rigorous Process

The adaptation of Mathiassen et al.'s formulation of a rigorous five-staged process of planning, development, preparation, experimentation and evaluation [Mathiassen et al., 1998] along with knowledge gained through Thorshøj's detailed description of experiments in systems development [Thorshøj, 1989], served the purpose of organizing our process well. The result was a very efficient experiment with the underlying system ideas of the agenda prototype and the war stories prototype as described in Sections 5.1 and 5.2. The clear definition of objectives and sub-objectives for the two iterations, subsections 5.1.1 and 5.2.1, served as guidance for the whole process. Hereby, we were able to perform the development stages very efficiently, using only approximately 80 man hours for the first iteration and approximately 32 man hours for the second iteration.

Another point concerns the coupling of a rigorous process with a rather loosely conducted experimentation where the evaluation of the prototypes took the form of debriefing, see sections 5.1.4 and 5.2.4. During the experimentation, the test person was free to speak his mind and try out features of the prototype. Via this coupling we were able to take advantage of our test person's great experience with systems development and with evaluating and testing systems. Due to his experience, he was able to abstract from the concrete implementation of the prototype in order to evaluate the underlying system ideas, see Subsection 5.1.4 and 5.1.5. In this way, our test person did not resolve to commenting specific details of the prototypes, but rather he was able to relate to how the prototype was going to be used and how it was going to affect the work of P^+.

Involving Test Persons

With respect to the selection of test persons, we view the use of only one test person as a minor setback. It would have been preferable to include more P^+ members in our experiments in order to provide a more thorough evaluation, possibly providing richer feedback and different perspectives on the two prototypes. As argued in Subsection 5.2.5, this would have enabled us to evaluate the prototypes as groupware and study the effects of a multi-user environment. Unfortunately, time did not allow us to conduct such experiments as the next P^+ meeting was scheduled after the hand-in and exam of this study. However, taking the arguments of Saarinen and Sääksjärvi into consideration [Saarinen and Sääksjärvi, 1990], as described in Subsection 2.4.5, it is the quality of involved users rather than the quantity that influences the success of an information systems development project. In this respect, our contact person

served our purposes well as a key member of P^+ and a highly committed project manager. Through his work, he is very much in touch with the interests of P^+ because he is a key contributor to the work of the group. As described in the previous paragraph, he was able to abstract from the specifics of the prototypes and evaluate the underlying system ideas. Therefore, we do not view our involvement of only one test person as compromising to our results, but rather it may be argued that we would have been able to provide a more thorough evaluation of the system ideas by involving other P^+ members.

Cooperativeness of the Prototyping Phase
The cooperative style of prototyping used in the experimentation and evaluation of the developed prototypes proved very useful as the purpose was to evaluate underlying system ideas of the prototypes. This is seen from both of our experimentation stages described in subsections 5.1.4 and 5.2.4. The freedom provided by the open-ended approach gave the test person an opportunity to explore the prototypes on his own without restrictive boundaries imposed by us. Such open-endedness seemed to fit the informal company culture at WM-data well. In particular, the test person seemed very comfortable with evaluating the prototypes and had no concerns about uttering his opinion, both with respect to the usefulness of the prototypes' underlying system ideas and with respect to specific details concerning the functionality, see subsections 5.1.4 and 5.2.4. Furthermore, the experimentation and evaluation of the prototypes formed the basis of a discussion, focusing on both system ideas and functionality. In this way, we were able to instantly evaluate ideas, which came up, and discuss potential improvements, e.g. the prioritization of agenda items, see Subsection 5.1.4.

Establishing a Shared Context

The experimentation and evaluation of the system ideas through the prototypes showed very valuable during the experimentation stages of our experiments, sections 5.1.4 and 5.2.4. The prototypes acted as a means of communication through which ideas and thoughts could easily be shared. In particular, the concreteness of the prototypes helped in understanding our test persons viewpoints and comments. As a consequence, a *ba* was established where knowledge was transferred back and forth between the test person and ourselves. This establishment was also facilitated by the fact that our test person was an experienced project manager (and former systems developer) and a key member of P^+ with whom we have had much prior contact. Furthermore, the fact that he was also our cooperative partner during the comparison stage of our SSM analysis, see Subsection 4.5.1, facilitated the establishment. Another attributing aspect was our similar educational backgrounds and focus on systems development. All of these aspects contributed to the fostering of a shared context.

From the point of view of the *shared language game* as proposed by Ehn [Ehn, 1989]

and described briefly in Subsection 2.4.5, the shared language was already partly in place so that communication could progress without problems. As Bondgård et al. point out, "understanding the language of a professional involves the ability to master the practical rules which are created through the social tradition of the profession — the ability to participate in language games" [Bondgård et al., 1990b, pp. 15]. Such participation was made possible through our *ba*. Combining the ideas of Ehn, Polanyi and Nonaka and Konno, the prototype became the communication medium through which the test person and ourselves were able to explicate ideas and viewpoints. It helped in setting up the shared language and the context in which knowledge could be acquired by both the test person and ourselves, leading to an evaluation of the two prototypes and their underlying system ideas.

6.2.3 Modelling the Combined Process

Our combined process of SSM and prototyping has through our application in the specific case of tailoring knowledge management systems to support the work of P^+, shown to be very useful. Figure 6.1 shows a conceptual model of our combination of SSM and prototyping.

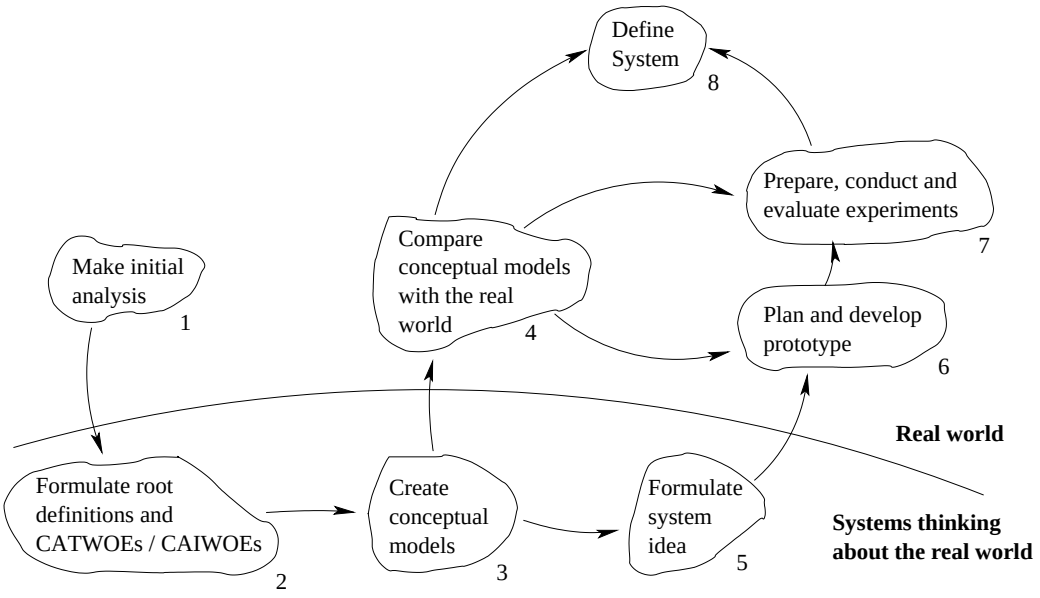


Figure 6.1: *Combination of SSM and prototyping.*

The model reflects our actual process of combining SSM and prototyping from the initial analysis (1) to the end-goal of defining a system to support the work of P^+ (8). It has evolved as a result of our experiences with working with SSM and prototyping. It is important to notice that the term *system* is ambiguous in the sense used here. The stage 'Define system' contains both the properties of the human activity

systems, i.e. the *use* of the knowledge management *system*, and the properties of the knowledge management application as expressed through the prototypes. For clarity and convenience, we have modelled the five stages of our prototyping process as two stages: One stage for planning and development (6) and one stage for preparation, conduction and evaluation of experiments (7). It should furthermore be noted that our use of both SSM and prototyping has been iterative both with respect to the use of SSM and our use of prototyping, but also in their combination. Such iterativeness cannot be inferred from the model, but we emphasize that the stages of the model can be chosen in any order suitable. For instance, the creation of conceptual models (3) affected the initial analysis (1) and the preparation, conduction and evaluation of experiments (7) lead to modifications to the system ideas (5).

We are not going to describe each stage of the model of Figure 6.1 because it reflects how we have applied SSM and prototyping during our study. Such details can be found in Chapters 4 and 5. Instead, we focus on the transitions between the SSM stages and the prototyping stages, in particular the combination of stages (3, 4, 5, 6 and 7) and how these stages lead to the definition of the system (8). This enables us to address four of the following results:

- Representing the prototypes' underlying system ideas through conceptual models ($3 \rightarrow 5$)
- Defining which prototypes to develop through comparison ($4 \rightarrow 6$)
- Utilizing conceptual models and root definitions in informal requirements specifications for prototypes ($4 \rightarrow 6$)
- Guiding the formulation of experimentation objectives through the conceptual models ($4 \rightarrow 6$)
- Utilizing the conceptual models in experimentation ($4 \rightarrow 7$)
- Defining the system ($4 + 7 \rightarrow 8$)

Representing the Prototypes' Underlying System Ideas through Conceptual Models ($3 \rightarrow 5$)

The conceptual models were found very useful in formulating system ideas for the prototypes, see Subsection 5.1.1. The system ideas reflected both the intended use of the system as expressed via the conceptual model and the knowledge management theory that was incorporated into the conceptual models. In this way, the conceptual models were concretized into system ideas. As a consequence, the system ideas reflected the activities of the conceptual models and served as the basic ideas for the prototypes.

Utilizing Conceptual Models and Root Definitions in Informal Requirements Specifications for Prototypes (4 → 6)

During the initial development of the prototypes, Subsection 5.1.2, the conceptual models were used to derive functionality. This proved very useful because it enabled us to tailor the prototypes according to the activities of the systems. One example concerned the ability to propose items for an agenda. This functionality was an activity of the conceptual model for the agenda system, see Subsection 4.4.2. The activity was supported directly via the prototype by including the functionality required to propose an item, see Subsection 5.1.2. In this way, a more tangible list of requirements was established, serving as an informal requirements specification. Thus, the conceptual models provided guidelines as to how the prototypes were to be implemented in order to support the system.

Utilizing the Conceptual Models in Experimentation (4 → 7)

The comparison stage lead to the selection of two systems which were to be experimented with using prototypes. This also influenced how the experiments were to be prepared and conducted. By using our cooperative partner from the comparison stage, we were able to base our experiment on a more loosely conducted experimentation and evaluation stage. This was mainly due to our test person's experience in systems development and his familiarity with the systems. Furthermore, we used the conceptual models to form an informal notion of relevant use cases. These use cases were, however, not made explicit, but rather served as an understanding of expected behaviour. Through our experiment, we observed whether the use cases implicitly suggested by the conceptual models corresponded to the test person's actual use of the prototype. This assisted in the validation of the system ideas, addressed in Section 6.3.

Defining the System (4+7 → 8)

The definition of the system served as the main result from our efforts of combining the SSM analysis, Chapter 4, and the prototyping phase, Chapter 5. The results from the prototyping phase were two executable specifications of two knowledge management applications that specified the properties of a future knowledge management application. The results achieved through the SSM analysis were two systems that modelled the knowledge activities surrounding the future knowledge management applications. By combining both results, we defined a set of properties for a knowledge management *system*, comprising both the system (in the sense of SSM) and the application. In effect, this resulted in a specification for a tailored knowledge management system which incorporates both the system to use the knowledge management application and the knowledge management application itself.

6.2.4 SSM and Prototyping

The iterative nature of both SSM and prototyping showed to be very suitable when working within the complex organizational context of WM-data. Vidgen and Braa's notion of 'the Organizational Laboratory' (see Section 2.1) described our setting well. As a result of the iterativeness of SSM, the human activity systems modelled relevant systems within P^+. In this way, SSM provided an abstract understanding of knowledge management activities that supported the work of P^+. Correspondingly, the iterative nature of prototyping resulted in the development of two prototypes that supported the human activity systems modelled through SSM. These prototypes were tailored to P^+ through two iterations. The correspondence is shown in Figure 6.2.

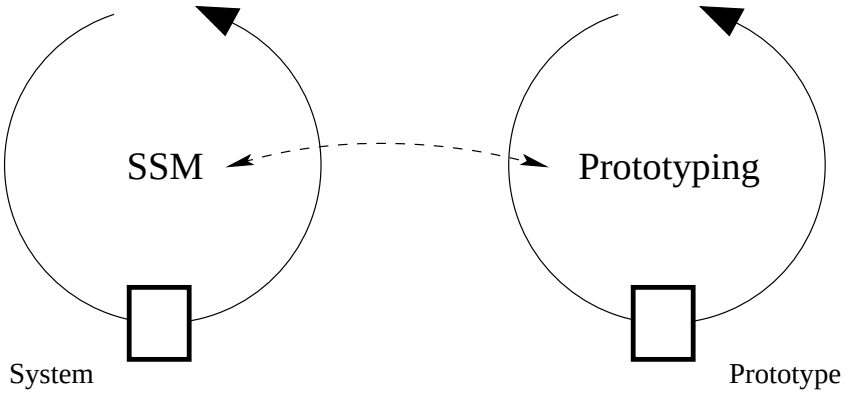


Figure 6.2: *Correspondence between SSM and prototyping.*

In this way, the iterativeness of SSM is similar to that of prototyping and the two methodologies have shown to support each other well in accordance with the 'Principle of Limited Reduction' [Mathiassen and Stage, 1992].

6.3 Usefulness of the System Ideas

By conducting the comparison stage, we ended up with two systems that suited some of the needs for knowledge management support within P^+. The underlying system ideas of the two prototypes, as expressed in Subsection 5.1.1, were concretized through two prototypes of future knowledge management applications. In the experiments with the prototypes, the test person expressed great satisfaction with the usefulness of the two system ideas based on the conceptual models. Our test person's accept of our analytical results together with his appreciation of our prototypes showed the usefulness of our systems. We thus consider the usefulness of our proposed

systems as a *proof-of-concept*.

In addition to delivering our *proof-of-concept*, we could have further validated the system ideas by including other P^+ members in our experiments and testing the prototypes as groupware, i.e. test the capabilities of the prototypes in a multi-user environment. Sadly, such an experiment could not be set up within the time frame of our research period.

6.3.1 Validation of System Ideas

The formulation of sub-objectives for each iteration of prototyping, in addition to the main objectives, served as a useful way of making sure that problematic aspects of the system ideas and the prototypes were addressed during the experimentation. In addition, this strengthened the *proof-of-concept* of our two system ideas. In the following, we address three overall aspects of the system ideas inspired by the sub-objectives of sections 5.1.1 and 5.2.1:

- Establishment of a shared context and understanding
- Fostering of commitment
- Codification of narratives

The establishment of a shared context and understanding was an important aspect of both system ideas. In the case of the agenda system, such establishment was to be facilitated by sharing and discussing agendas prior to meetings. Following Nonaka and Konno, the system idea of the agenda prototype should facilitate the creation of *cyber ba*, see Subsection 3.1.2, where explicit knowledge were to be *combined* to form new knowledge. This *ba* was thought to be relevant because the P^+ members do not assemble very often. The experiments showed this theoretical understanding of the underlying system idea to be relevant. In the opinion of the test person, the discussion of agenda items would add to the shared context prior to meetings. However, it was emphasized that the most important part of the establishment would still be achieved through personal interaction among group members. In the case of the system idea for the war stories system, it was stated that it was the telling of the story that would add to the establishment of a *ba*, not so much the retrieval of the story via the prototype.

The fostering of commitment was supported via both system ideas. The test person noticed that the sharing of agendas prior to meetings, in his opinion, would lead to increased commitment and more focused meetings because participants would likely have formed an opinion about topics and issues in advance. In the case of the underlying system idea for the war stories system, such commitment would likely be fostered via the telling of the story and the quest for relevant stories.

With respect to the codification of narratives (exemplified through war stories), the experiments showed that the structural format should be tailored to fit the characteristics of P^+. The addition of a 'lessons learned' field and a 'keywords' field seemed to improve the codification, but the fields of 'what', 'how' and 'why' would require more experimentation; the semantics of these fields has to be understood by everyone (Subsection 5.2.4). With respect to the context of the story, the attributes proved very useful because they put focus on relevant characteristics of the story such as teller, location of incident etc. In this way, part of the war story's context could be fostered. This could, however, never replace the actual telling of the story. Other highly relevant aspects of the story's context such as body language of the teller, facial expressions and tone of voice were found difficult to codify. This would require a separate study and would be an interesting topic for future work.

6.3.2 Sizing up the System Ideas

With respect to the size and scale of the prototypes, it is interesting to notice that the two prototypes represented rather simple system ideas. This suggests that it is not the comprehensiveness of the prototypes that determines their usefulness, but rather the way they are to be used as described by the human activity systems from SSM. For instance, the agenda prototype represents a very simple idea about cooperative creation of group agendas. The prototype is very simple, but still the system idea would serve its purpose well in the opinion of the test person, see Subsection 5.1.4. This is also the case for the underlying system idea of the war stories prototype. One reason could be that the usefulness of simple system ideas is closely related to the informal company culture of WM-data. Such informal culture also has its impact on P^+ and their initiatives. It may therefore be the case that the work of P^+ is best supported by rather simple applications that are tailored to support their main operation: Sharing knowledge between projects. In this way, their informal practices are not compromised by large applications where too much time is spent on formalities. This also corresponds to the fact that our research from our previous study showed that a personalization strategy would be most suitable in WM-data. Although the two human activity systems of the agenda system and the war stories system, see subsections 4.4.2 and 4.4.1, respectively, were based on a codification strategy, the main focus is still on how they are used. Neither of the two prototypes are intended to be used as replacements for personal face-to-face communication, substituting human interaction; they are rather meant for support. It may therefore be argued that Hansen's proposal of a 80-20 split between personalization and codification in a personalization strategy has not been compromised; the 80% personalization is addressed via the human activity systems.

This leads to one final observation. The above reasoning may suggest that when following a personalization strategy there is a less need for large-scaled knowledge management systems. Instead, minor knowledge management systems may prove useful because these do not compromise the person-to-person interaction charac-

terizing a personalization strategy. By utilizing large-scaled knowledge management systems, the system becomes the focus of attention, not the people.

6.4 On The Edge of Hard and Soft Systems Thinking

This section winds up the results of our study and provides a perspective on our difficulties of adapting soft systems thinking. Our intention is to view these difficulties on a broader scale by emphasizing the troublesome process that we have undergone as hard systems thinkers. We hope that our thoughts will assist in providing a better understanding of what makes this process so difficult.

The following questions represent the topics which are addressed in the following paragraphs:

> *What is it that makes the adoption and application of SSM so complex? and Why is SSM traditionally not applied in software organizations which develop knowledge management systems?*

6.4.1 Hard Systems Thinkers and SSM

To address the first question, it may be argued that our general problem of applying SSM strongly relates to our educational background. The educational programme of computer science is based on natural science which indeed represents hard systems thinking. Courses such as mathematics and systems analysis, which are part of the educational programme of computer science, both represent courses in which reductionism, i.e. reducing the complexity of a problem into smaller and less complicated parts, is applied in reaching one true solution. Therefore, our educational background has been one of the obstacles in our attempt to adopt soft systems thinking. Furthermore, the idea of working concurrently with more than one perspective on how to solve the same problem and treating these as equally important seems peculiar to hard systems thinkers. Such thinkers will unconsciously try to reduce the complexity of a problem via decomposition. We believe this perception to be common to many others. As hard systems thinkers, we therefore, often without noticing it, discard alternative perspectives even before the plausibility of these have been fully evaluated. Therefore, the unconscious processes have in addition served as one of the obstacles in our adoption of soft systems thinking during this study.

To elaborate on the second question, we take these considerations to the next level by considering the application of SSM in software organizations. We believe that due computer scientists' application of traditional hard systems thinking, this has a great impact on the techniques adopted by software organizations for analyzing problematic situations. Another aspect is the evolution of society from an information society towards a knowledge society. In the information society, the primary

focus is on developing cost-effective software. That is, software is developed to cut down expenses of e.g. administrative tasks such as keeping track of students, their grades and passed courses. Common for such software is the transformation of a tangible input to a defined output through the transformation process delivered by the software.

However, as we now progress towards a knowledge society, we believe that the requirements for software have changed. There will still be a demand for software of the same type as in the information society, but another interesting aspect is receiving a great deal of attention, namely knowledge management systems. To stay competitive in todays ever changing business markets, software organizations have to focus on how to manage existing knowledge and how to create new knowledge. A lot of unsuccessful attempts have already been made at designing, developing and implementing knowledge management systems as stated in the introduction, Chapter 1. Why is that so? We believe that the reason lies within the characteristics of knowledge. Knowledge resides in the human mind and is based on a personal justified true belief as defined by Nonaka [Nonaka, 1994]. Defining the transformation process from input to output for a knowledge management system can therefore be difficult due to the intangible nature of knowledge. Consequently, human activity systems related to the management of knowledge may be better visualized and expressed via interaction systems because a direct transformation into knowledge cannot be quantitatively measured, see Section 4.4.

Moreover, due to the intangible nature of knowledge, the fostering of knowledge sharing and knowledge creation is dependent on human interaction and, according to Nonaka and Konno, whether or not the proper *ba* is present [Nonaka and Konno, 1998]. In this sense, *ba* can be compared to the surrounding environment in which sharing and creation of knowledge are to take place. We have found that the surrounding environment evolves continuously through time and is affected by factors such as company culture, communication style, management style, cooperation style among co-workers, organizational structure and team spirit. All of these factors have an impact on how knowledge is shared and created. So in order to analyze a problem situation related to management of knowledge, we must not only analyze the situation in question, but also the surrounding environment. Consequently, the various considerations lead to different perspectives on how to improve the problem situation. By considering various perspectives, we circumvent the core belief of hard systems thinking, namely that only one true representation exists. Accordingly, we propose to adopt SSM as the primary means of analysis within knowledge management.

However, SSM does not specify how the knowledge applications, supporting the human activity systems are to be designed and implemented. These aspects are addressed by prototyping. We have found that by initiating prototyping based on selected systems, we have been able to tailor the knowledge management application to the actual demands of an organization, in our case WM-data. Furthermore, the

iterativeness of prototyping ensures that we are able to tailor and refine the prototypes to the changing needs and demands of the organization and the surfacing of new ideas.

Finally, we propose that when tailoring knowledge management applications, soft systems thinking should to a greater extent be applied due to the complex nature of knowledge. In addition, in order to evaluate the usefulness of the system ideas concretize these ideas, prototyping can be used for defining the knowledge management application as a means of visualizing the first steps towards its design and implementation. Based on these considerations, we propose a combination of SSM and prototyping when tailoring knowledge management systems.

Chapter 7

Conclusion

This study was initiated due to an interest in the field of knowledge management and in the tailoring of knowledge management systems, in particular.

The knowledge management literature contains numerous reports on failed attempts at developing and implementing knowledge management systems. Many of the systems are never put to use because they are not suitable for the organization. It may be argued that the main reason for these unsuccessful implementations of knowledge management systems lies within a misconception on how software organizations attack and manage knowledge related-problems. Often, analysts in software organizations are products of the traditional school of hard systems thinking due to their educational background, Section 6.4. It may be argued that it is this traditional approach of solving problems that represents the core obstacle for successfully tailoring knowledge management systems. The following question has served as the theme of this study:

> **Research question:**
> *How can the process of tailoring knowledge management systems benefit from using Soft Systems Methodology and prototyping?*

In order to explore our research question, we have studied literature within Soft Systems Methodology (SSM) and prototyping. We note that through our previous study we gained an in-depth understanding of the theory within the field of knowledge management which we also have applied in the current study.

The process of this study has been based on an action case approach, a hybrid research approach between action research and case studies, see Section 2.2. In order to address the very intangible nature of knowledge-related problems, we have applied SSM as it enables us to consider several perspectives of a problem situation. This is required because aspects such as company culture, communication style and management style all have an impact on how knowledge is shared and created. These

issues must also be considered in order to fully understand the problematic situation. This also represents the main reason for choosing SSM as our tool for analysis. In addition, we found it necessary to make the analytical findings, i.e. human activity systems, concrete. In this respect, we found prototyping to be useful due to its experimental approach. By combining these two methodologies, we combine an analytical mode of operation and an experimental mode of operation in which SSM and prototyping compensate for each others weaknesses. Consequently, our combination of approaches satisfies Mathiassen and Stage's 'Principle of Limited Reduction' [Mathiassen and Stage, 1992], see Section 1.2.

7.1 Contributions

Our contributions due to this study are very much affected by our experiences and relate both to the actual application of SSM and prototyping, but also to the combination of the two methodologies. In the following subsections, we present, in our opinion, the most interesting results of our study. For an in-depth explanation of all our results, we refer to Chapter 6.

7.1.1 SSM

With respect to our application of SSM, we found that the adaptive property of systems in SSM is very suitable when analyzing the ever changing characteristics of knowledge. Through our work, we found it necessary to adapt SSM to our specific needs which resulted in the following contributions.

- Adding Analysis Four to the cultural enquiry
- Use of interaction systems

Through our previous study, [Hosbond and Ørtoft, 2003], we gained a thorough understanding on the types of knowledge, knowledge levels and how knowledge in general was managed in WM-data. It may be argued that in order to improve how knowledge is managed in an organization, an understanding on how knowledge currently is being managed is required. We therefore add a fourth dimension to the cultural enquiry based on the findings of our previous study. Furthermore, it is difficult to express systems which are characterized by the main activity of managing something. This was experienced through our work with the logic-based enquiry described in Chapter 4. In effect, this made the construction of conceptual models problematic. The consequence was that we experienced difficulties defining a tangible input and output as required in a traditional transformation system. We therefore adopted interaction systems, proposed by Mathiassen and Nielsen

[Mathiassen and Nielsen, 1998], which are systems characterized by an invariant state space, see Subsection 2.3.5.

7.1.2 Prototyping

Through our process, we acknowledged the strengths of prototypes. By developing prototypes, we were able to concretize the abstract representation of systems from SSM. Although the prototypes did not fully represented the human activity system, they still provided a means of visualizing the underlying system idea so that this could be evaluated by the test person. Consequently, the evaluation of the experiments served as a way of validating the usefulness of the system ideas, providing us with a *proof-of-concept*. Through our experiences with prototyping, we especially found one aspect intersting, namely that of:

- Establishing a shared context

Through the experimentation and evaluation stages of each prototype, we found that the prototype acted as a means of communication through which ideas and thoughts could easily be shared between the test person and ourselves. Consequently, a shared context (*ba*) was established where knowledge was transferred back and forth between the test person and ourselves. The establishment of *ba* was facilitated by our prior contact with the test person. Furthermore, the test person had a similar educational background, making the establishment of *ba* even less problematic. The shared context increased the amount and quality of relevant feedback which, in effect, lead to a more tailored refinement of the prototypes.

7.1.3 Combining SSM and Prototyping

Our main contribution lies within the methodological combination of SSM and prototyping. In this respect several experiences were encountered:

- A model for tailoring knowledge management systems
- Analytical results from SSM concretized through prototyping

Through the combination, we developed a model based on SSM and prototyping. The model is a result of our work and represents the actual process describing our process of tailoring knowledge management systems. The model is shown in Figure 7.1.

It has been argued that the model benefits from the analytical strength of SSM and the experimental strength of prototyping. Furthermore, we have found that the model is capable of adapting to changing requirements due to the iterative nature

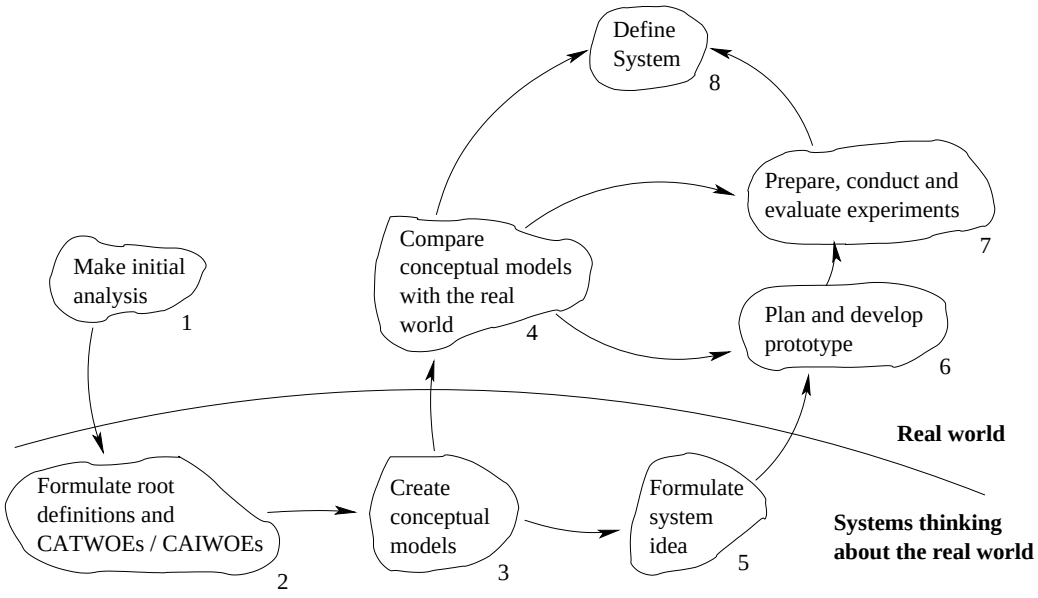


Figure 7.1: *Our proposed model for tailoring knowledge management systems. A combination of SSM and prototyping.*

of both SSM and prototyping. In addition, we emphasize that the use of our model is up to the problem solver (analyst, developer or both). That is, the stages of the model can be iterated in any order required. In effect, this in general makes the model adaptable to the various needs of a problem solver.

The following experiences relate to our proposed model in Figure 7.1.

- Utilizing Conceptual Models and Root Definitions in Informal Requirements Specifications for Prototypes ($4 \rightarrow 6$)
- Utilizing the Conceptual Models in Experimentation ($4 \rightarrow 7$)
- Defining the System ($4{+}7 \rightarrow 8$)

Through our work with P^+, we experienced how SSM and prototyping actually combined. In our use of the model, we combined SSM and prototyping through the comparison stage (4). We have experienced no problems due to the combination. In addition, we have found that the underlying system ideas of the prototypes represent the conceptual models of the selected systems which together with the root definitions act as informal requirements specifications for the prototypes.

Furthermore, we have found that the conceptual models, which express the activities of the systems and their intended interrelations, can indeed be used to formulate objectives for experiments using prototypes. That is, we may want to test whether

these interrelations between actions in the conceptual model are actually supported by the prototype.

The last account relates to stage 8 ('Define system') in the model. It may be argued that the stage combines the overall results of both SSM and prototyping. The results from SSM express the entire system (e.g. activities and their interrelations, actors in the system, constraints on the system etc.) in an abstract manner. On the other hand, the prototype represents a knowledge management application capable of supporting and improving the activities of the system. Consequently, it is argued that the 'Define system' stage comprises both the analytical results and the experimental results which together lay the foundation for the development for a future knowledge management system.

7.2 Future Work

During our research, several interesting issues have arisen that may be subject to future work. First, the applicability of our model needs to be experimented with during future applications. Our study has only involved one case, namely the P^+ initiative of the Systems Development branch of WM-data Denmark. It would be interesting to see, if the model may be used in other situations. Second, the step from stage 8 ('Define system') to the actual development and deployment of a knowledge management system would also be interesting to investigate. For instance: How may the human activity systems be incorporated into existing practices?

Third, it would be interesting to further explore the possibilities of codifying war stories. Is it at all possible to adapt the structural format to the characteristics of a company? and How may this affect the presentation of the war story and the impact made on the reader/listener? Finally, the potential of utilizing conceptual models in the generation of use cases/user stories would provide a way of incorporating the results of the SSM analysis into more traditional systems development methods such as object-oriented analysis and design, rapid application development or extreme programming. This would pave the way for even more effective knowledge management systems.

Bibliography

[Basili et al., 1994] Basili, V., Caldiera, G., and Rombach, H. (1994). The Experience Factory. *Encyclopaedia of software engineering*, pages 528–532. Wiley.

[Baskerville and Wood-Harper, 1998] Baskerville, R. and Wood-Harper, A. (1998). Diversity in information systems action research methods. *European Journal of Information Systems*, Vol. 7:90–107.

[Blum, 1955] Blum, F. H. (1955). Action Research – A Scientific Approach? *Philosophy of Science*, Vol. 22(1):1–7.

[Bondgård et al., 1990a] Bondgård, P., Degn, E., and Vraaggaard, K. (1990a). Designing Effective Prototyping Experiments. Aalborg University.

[Bondgård et al., 1990b] Bondgård, P., Degn, E., and Vraaggaard, K. (1990b). The Gap Between Prototyping Theory and Practice. Aalborg University.

[Braa and Vidgen, 1997] Braa, K. and Vidgen, R. (1997). An IS Research Framework for the Organization as Laboratory. In Kyng, M. and Mathiassen, L., editors, *Computers in Context: Joining Forces in Design*. Cambridge, Massachusetts: MIT Press (forthcoming).

[Brown and Duguid, 1991] Brown, J. S. and Duguid, P. (1991). Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation. *Organization Science*, Vol. 2(1):40–57.

[Budde et al., 1992] Budde, R., Kautz, K., Kuhlenkamp, K., and Züllighoven, H. (1992). What is prototyping? *Information, Technology & People*, Vol. 6(2):89–95.

[Checkland, 1975] Checkland, P. B. (1975). The development of systems thinking by systems practice — a methodology from an action research programme. In Trappl, R. and Hanika, F. d. P., editors, *Progress in Cybernetics and Systems Research*, volume Vol. II. Hemisphere Publications, Washington.

[Checkland, 1981] Checkland, P. B. (1981). *Systems Thinking, Systems Practice*. John Wiley & Sons, Chichester.

[Checkland and Scholes, 1990] Checkland, P. B. and Scholes, J. (1990). *Soft Systems Methodology in Action*. John Wiley & Sons. ISBN: 0-471-92768-6.

[Creswell, 1994] Creswell, J. W. (1994). *Research Design: Qualitative & Quantitative Approaches*. Sage Publications, Inc. ISBN: 0-8039-5255-4.

[Dahlbom and Mathiassen, 1995] Dahlbom, B. and Mathiassen, L. (1995). *Computers in Context*. Blackwell Publishers. ISBN: 1-55786-405-5.

[Ehn, 1989] Ehn, P. (1989). Playing the language games of design and use. In *Proceedings of COIS '88*. Palo Alto.

[Floyd, 1984] Floyd, C. (1984). A systematic look at prototyping. In Budde et al. (eds) *Approaches to Prototyping*, Springer-Verlag, pp. 1-19.

[Grønbæk, 1989] Grønbæk, K. (1989). Extending the Boundaries of Prototyping - Towards Cooperative Prototyping. In Bødker, S., editor, *Proceedings of the 12th IRIS conference*, pages 219–239. DAIMI PB-296-I. Aarhus, Denmark: Aarhus University, Computer Science Department.

[Hansen et al., 1999] Hansen, M. T., Nohria, N., and Tierney, T. (1999). What's Your Strategy for Managing Knowledge. *Harvard Business Review*, Vol. 77(2):106–118.

[Hellström et al., 2001] Hellström, T., Malmquist, U., and Mikaelsson, J. (2001). Decentralizing knowledge: Managing knowledge work in a software engineering firm. *Journal of High Technology Management Research*, Vol. 12(3).

[Hosbond and Ørtoft, 2003] Hosbond, J. H. and Ørtoft, R. (2003). Knowledge Management: An Empirical Investigation of Software Process Knowledge. Aalborg University. Supervisor: Peter A. Nielsen.

[Hult and Lenning, 1980] Hult, M. and Lenning, S. (1980). Towards a definition of action research: A note and bibliography. *Journal of Management Studies*, 17(May):241–250.

[Iversen et al., 2002] Iversen, J., Mathiassen, L., and Nielsen, P. A. (2002). Risk Management in Process Action Teams. In Mathiassen, L., Pries-Heje, J., and Ngwenyama, O., editors, *Improving Software Organizations: From Principles to Practices*, chapter 16, pages 273–285. Addison-Wesley. ISBN: 0-201-75820-2.

[Kautz and Thaysen, 2001] Kautz, K. and Thaysen, K. (2001). Knowledge, learning and IT support in a small software company. *Journal of Knowledge Management*, Vol. 5(4):349–357.

[Lewin, 1947] Lewin, K. (1947). Frontiers in group dynamics II. *Human Relations*, I(2):143–153.

[Madsen, 2002] Madsen, F. P. (2002). *Cookbook on Startup Seminars*. WM-data.

[Mathiassen, 1998] Mathiassen, L. (1998). Reflective Systems Development. Dr. Techn. Thesis, Aalborg University.

[Mathiassen et al., 1998] Mathiassen, L., Munk-Madsen, A., Nielsen, P. A., and Stage, J. (1998). *Objektorienteret analyse og design*. Forlaget Marko Aps, Aalborg, 2. edition. ISBN: 87-7751-129-8.

[Mathiassen and Nielsen, 1998] Mathiassen, L. and Nielsen, P. A. (1998). Interaction and Transformation. In Mathiassen, L., editor, *Reflective Systems Development*, chapter 13, pages 329–345. Dr. Techn. Thesis, Aalborg Univerity.

[Mathiassen et al., 2003] Mathiassen, L., Robertson, M., and Swan, J. (2003). Cracking the Code: The Dynamics of Professional Knowledge. Submitted to European Knowledge Management Conference.

[Mathiassen and Stage, 1992] Mathiassen, L. and Stage, J. (1992). The principle of limited reduction in software design. *Information Technology and People*, Vol. 6(2).

[McGrath, 1982] McGrath, J. (1982). Dilemmatics: The Study of Research Choices and Dilemmas. In McGrath, J., Martin, J., and Kulka, R., editors, *Judgement Calls in Research*, pages 69–102. Beverly Hills: Sage.

[Nonaka, 1994] Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organizational Science*, Vol. 5(1):14–37.

[Nonaka and Konno, 1998] Nonaka, I. and Konno, N. (1998). The Concept of "Ba": Building a Foundation for Knowledge Creation. *California Management Review*, Vol. 40(3).

[Pfleger, 2001] Pfleger, S. L. (2001). *Software Engineering Theory and Practice*. Prentice Hall, 2nd edition. ISBN: 0130290491.

[Polanyi, 1962] Polanyi, M. (1962). *Personal Knowledge: Towards a Post-Critical Philosophy*. Routledge & Kegan Paul Ltd. Reprinted in 1998, ISBN: 0-415-15149-X.

[Pressman, 2000] Pressman, R. S. (2000). *Software Engineering: A Practitioner's Approach (European Adaptation)*. McGraw-Hill Publishing Company, fifth edition, ISBN: 0-07-709677-0.

[Saarinen and Sääksjärvi, 1990] Saarinen, T. and Sääksjärvi, M. (1990). The Missing Concepts of User Participation: An Empirical Assessment of User Participation and Information System Success. *Scandinavian Journal of Information Systems*, Vol. 2:25–42.

[Schultze and Boland, 2000] Schultze, U. and Boland, R. (2000). Knowledge management technology and the reproduction of knowledge work practices. *Journal of Strategic Information Systems*, Vol. 9:193–212.

[Swan et al., 1999] Swan, J., Newell, S., Scarbrough, H., and Hislop, D. (1999). Knowledge management and innovation: networks and networking. *Journal of Knowledge Management*, Vol. 3(4):262–275.

[Thorshøj, 1989] Thorshøj, K. (1989). *Eksperimenter i systemudvikling.* Roskilde Universitetscenter. Found at "FTU Boghandlen, Halmstadgade 6, 8200 Aarhus N".

[Trigg et al., 1991] Trigg, R. H., Bødker, S., and Grønbæk, K. (1991). Open-ended Interaction in Cooperative Prototyping: A Video-Based Analysis. *Scandinavian Journal of Information Systems*, Vol. 3:63–86.

[Vidgen and Braa, 1997] Vidgen, R. and Braa, K. (1997). Balancing Interpretation and Intervention in Information Systems Research: The Action Case Approach. In Lee, A. S., Liebenau, J., and DeGross, J. I., editors, *Information Systems and Qualitative Research*, chapter 26. Chapman & Hall. ISBN: 0-412-82360-8.

[Zack, 1999] Zack, M. H. (1999). Developing a Knowledge Strategy. *California Management Review*, Vol. 41:25–38.