Coexistence or no existence:

ERP systems as process protocols

Ph.D Thesis
by Lene Pries-Heje
December 2008

Preface

When you have worked for many years with implementation of ERP systems you know that there is more to it than just following an implementation guide. But rarely do you find an hour or a day where you can think more deeply about what this 'more' could be.

It was therefore a great pleasure when my application for a KMD-financed Ph.D. was granted to me in 2005. So let me first of all express my sincere thanks to KMD and to the IT University of Copenhagen, Vice Chancellor Mads Tofte, who together created the opportunity of a 3-year Ph.D. scholarship that I took advantage of.

To get full and uncovered access to an actual ERP implementation I had to use my personal network. The godmother to my oldest son offered an opportunity to come into the organization that in this dissertation has been called Alfa. I will forever be indebted to her for this opportunity. And I am indeed very thankful to all the people from Alfa who used their time and shared their thinking with me. I cannot mention their names here – not to reveal the anonymous 'Alfa' – but they all deserve my warmest thank you.

It was also my personal network and their network that I used to gather the participants for the focus group discussions on ERP implementation that I organized. As you can see if you read on in this thesis the discussions in these focus groups have had an immense influence on my thinking and my contribution. My sincere thanks go to all the focus group participants.

The person that has influenced me the most in the 3.5 year period that this thesis has taken to be brought to life is my supervisor Yvonne Dittrich. There shall be no doubt that this dissertation would have looked very different if it had not been for Yvonne. We have had many discussions that have changed my perceptions and broadened my view. It was my lucky day when Yvonne was appointed to be my Ph.D. supervisor. Thanks for doing a great job, Yvonne.

Finally my thanks go to John Venable of Curtin University of Technology who made a half year visit to Western Australia possible. And to my family who have endured my total occupation of the ground floor of our house as working space, and my husband who have helped me with the layout of the thesis.

December 2008
Lene Pries-Heje.

Abstract

The scope of this thesis is the organizational implementation of ERP-systems. Based on existing literature that is dominated by a management perspective and by implementation guides from vendors I formulate my research question: How do organizations engage ERP implementation, and why does it often result in misfits?

To answer this question I undertake an empirical study in a large Danish organization. The study reveals that ERP implementation can be seen from two different angles that I call process and semantic, where process is about the coordination and mechanisms that together form a protocol, and semantic is about fulfillment of the need for standardized data. I also derive a categorization of misfits with four main types and a number of subtypes. And I specify how the knowledge needed is obtained in very different ways throughout the different phases of an ERP implementation.

Next I gather two focus group of external ERP experts and internal ERP experts (from user organizations) to talk about ERP implementation. My analysis reveals that ERP implementation can be perceived as 'a standardization war', as 'a game, and as 'a change project'. Then I link the knowledge needs from my case study with these 3 perceived metaphors of ERP implementation.

In the focus groups I also discuss which design artifacts they use and why? My answer is a number of techniques that is always useful, and some that can be useful, as well as techniques that may be useful.

My research methodology is what I have called truly hermeneutic and interpretive. In the concrete I present it in the thesis as going through four learning cycles where I (re-)interpret my data. I use a multi-method approach in that I change my way of collecting and/or analyzing data in each learning cycle. However, I only apply research methods and techniques that fit an interpretive perspective.

The outcome of the fourth learning cycle is a theory of ERP articulation work. This theory includes the process and semantic perspective as well as three layers of articulation called situated, local and federal.

The research in the thesis contributes to ERP research in different ways. First it contributes by adding rich insights about how organizations actually engage ERP implementation, and the ERP professionals' rationale behind different approaches. Second it contributes a new way to conceptualize ERP implementations as a socio-technical design that requires configuration and customization of the ERP software, but also articulation (meta-) work at different levels. Third it contributes by discussing how different perceptions (metaphors) of the ERP implementation process recognize and accommodate the articulation and metawork needed.

Content

Part I - Background and motivation

Part (I) *Background and motivation* contains three chapters that together are providing the basis for my research. The motivation for the research, the research questions, and the structure of the thesis is presented in chapter 1, the research method in chapter 2, and an overview of the ERP literature in chapter 3.

.

Chapter 1
Introduction & Research Question

This chapter sets the stage for my Ph.D. thesis. First I introduce ERP systems and look at the state-of-practice. Then I shortly take a look at state-of-the-art research, and identify a lack of human- and people-oriented research that takes the way organizations engage ERP implementation in-depth seriously. This then leads to the formulation of my research question. Finally I lay out the structure for the remainder of the dissertation.

1.1 Motivation

For more than four decades organizations have been engaged in developing and implementing Information Systems, and much research has been conducted in order to understand how to develop systems tailored to a specific organization. During the 1990s it however became clear that a different practice had slowly emerged in the area of Enterprise Systems Software. Thus over the years it had became more and more common for software providers to reuse existing code (code developed for prior customer) whenever developing an Information System for a new customer. At the end of 1990s this "reuse" of code was so extensive that Enterprise Systems Software very seldom was developed for an individual customer; it had become a packaged software product provided by different vendors such as SAP, Oracle, PeopleSoft, Baan and many more. Thus most Enterprise Software had become pre-developed software targeted at specific lines of businesses instead of tailor-made software for a specific customer. This phenomenon can be observed for different kinds of software e.g. Enterprise Resource Planning (ERP) software, Supply Chain Management (SCM) software, and Project Management (PM). In this thesis I will focus on ERP package software.

Typically ERP package software consists of a number of modules each addressing a specific functional area within an organization e.g. finance, warehousing, purchase, sales, human resource, or project management, and all the modules are integrated and share a common database. As a product Enterprise Package Software is marketed as impounding "best practice" work processes; developed based on knowledge about patterns of organizational work processes accumulated by the vendor over many years as a side effect of implementing software in a very large number of customer organizations. However, "best practice" does not necessarily mean that only one version of a process is supported by the software, in many functional areas alternative patterns are provided e.g. to accommodate different lines of businesses or different practices experienced within a specific area. Thus ERP package software may include thousands of business processes intended for a large variety of organizations, and instead of "best practice" it may be more appropriate to regard it as a collection of established patterns of organizational work processes.

It is very common to see the term "Standard (ERP) software" used as an alternative label for "ERP package software". Referring to ERP package software as standard software may however give a somewhat distorted impression of the software product and what is involved in setting it up and make it work for a specific organization. Thus ERP package software includes a large number (thousands) of configuration tables and application data tables that

needs to be populated with organizational data in order to make it useful. Hence ERP package software is generic "semi-finished" software that must be tailored to reflect the needs of a specific organization. Setting up the system deciding on values in the configuration tables and application tables is often referred to as *configuration* of the software (Bancroft 1998).

As mentioned above the ERP package software can be regarded as a collection of established patterns of organizational work processes. During configuration the organization have to decide which process patterns to use in their specific implementation of the software, and decide on detailed properties of the process patterns. Detailed decision may involve e.g. deciding if a process flows should require a delivery note to be created with or without a forced reference to a preceding order, or deciding if and how to split part of a process flow depending on the amount of the sales order. In order to perform the configuration a thorough understanding of the capabilities of the ERP package software is needed, and to decide on the configuration options a thorough understanding of the organizational processes is necessary. Understanding how to configure the ERP software is a rather challenging task, by some research referred to as a configuration knowledge barrier that organizations have to overcome (2002).

In some areas the ERP package software may not be able to accommodate indispensable organizational requirements, and in that case a *customization* of the software may be needed. By customization I understand a situation where functionality is added or altered by other means than configuration, typically customizing the application code. The amount and proportions of customizations may vary between implementations (Brehm, Heinzl et al. 2000), but most ERP implementations require some degree of customization.

Customizations are however considered risky when implementing ERP package software. First it may in itself be difficult to understand how to introduce changes into a huge integrated complex of application code, and it often pose a risk disrupting "standard" functionality somewhere else in the complex without knowing it. Second it may increase the costs and complexity upgrading to new releases of the software. Therefore part of the rational implementing ERP package software is to adapt the organizational work processes to the process patterns supported by the software instead of customizing the software.

Hence for the implementing organization it is extremely important to understand what capabilities configuration of the software provides, and then consider if any of the options is in harmony with existing/desired organizational practice. If not it is necessary to decide if the organizational practice should be changed or a customization should be performed. In the introduction to the book "Second-wave Enterprise Resource Planning Systems" (Shanks, Seddon et al. 2003) the challenge of implementing ERP package software is formulated this way:

> *"Apart from all the normal problems of information system project management, the novel difficulty for teams implementing Enterprise Software Systems is to decide which mix of configuration, customization, and process change is best for the organization."*

Establishing a sound base for deciding on the mix of customizations and organizational change may however be very challenging. It may be difficult to establish an understanding of the existing/desired organizational practice; the implementation often involves almost all

areas of an organization, and many stakeholders with very different needs and preferences. So how do organizations (implementation teams) arrive at an - at least partly shared - understanding of the existing/future practice? Considering the size and the complexity of the process complex provided by the ERP package software it may also be very challenging to develop a good understanding of the capabilities of the package software. How is such an understanding actually developed? Finally, is it at all possible for the implementation team to judge which processes will be best for the organization? The challenge seems to be understood but how can organizations actually go about establishing the necessary knowledge base for the decisions they have to make?

Research in the area of ERP implementations picked up very late in the 1990s. Thus as a research area ERP is fairly young, however two interesting literature reviews have already been published (Esteves and Pastor 2001) and (Moon 2007). Both reviews depict ERP research as a fairly immature and very diverse research area. A closer look at the papers included in the two literature reviews reveals that much of the ERP research with IS relevance is about ERP-project management, critical success factors implementing ERP, how to measure and ensure benefits of ERP implementations, and how to perform change management. Most of the existing ERP research has a (project) management perspective. I found only very few articles taking an interest in how implementation teams are composed and what kind of knowledge should be present. None of them actually investigates how design teams obtain the knowledge needed to decide on the mix of configuration, customization and organizational change.

Besides academic papers many books about ERP implementation have been written by ERP practitioners. Most of them explain how to implement ERP software provided by a specific vendor, e.g. (Cunningham, Dean et al. 2001), (Khan 2002), and (Bancroft, Seip et al. 1998). In these books re-engineering of organizational business processes seems to be a mantra. Thus it is recommended to re-engineer business processes prior to and/or during the ERP implementation. The main argument is that the implementation team will have to make decisions that will affect the organization for years to come, and in case re-engineering is postponed until after go-live then the implementation team will have to make decisions without a structured process for doing so ((Bancroft, Seip et al. 1998), page 125). For a successful implementation this literature understands change management to be the single most important critical success factor, I assume this is caused by the focus on re-engineering business processes.

Thus re-engineering business processes and managing organizational change (the management view) has received overwhelming attention from practitioners as well as researchers. It is anticipated that re-engineered organizational process; “to-be” processes, can be used as a specification for the configuration of the system, and that the necessary organizational changes can be concluded from the “to-be” processes. But do abstract business process descriptions and business cases provide an adequate basis for deciding on the design of the IT artifact? – I doubt it! And do implementation teams actually work as vendors and practitioner books recommend? – my experience tells me that is not the case!

Anyway, something seems to be wrong; empirical studies reveal that many companies experience moderate to severe business disruptions when going live with ERP package software, and they have difficulties recovering. Sometimes the only way to stabilize the

situation is to permanently increase staffing and reduce efficiency expectations. Furthermore it seems as if a turbulent ride after go-live results in a long lasting disinclination toward the ERP system ((Markus, Axline et al. 2003); (Pries-Heje 2008)). A hostile attitude toward the new ERP system makes it very difficult to improve the situation without conceding the user organization changes to the IT artifact. Often ERP experts argue that from a logical and technical point of view such changes may seem unnecessary, but in practice they may be inevitable in order to change the user organizations attitude toward the system. These difficulties imply that the socio-technical design plays an important role in the difficulties that organizations experience. A socio-technical perspective however seems to be almost absent in the existing ERP literature; it is more or less anticipated that the difficulties experienced is caused by resistance to change.

Organizations obviously have difficulties finding a way to make sound decisions about configuration, customizations and organizational changes that provide a useful socio-technical design. The ERP literature provides very few insights into how the design team and the organization actually engage ERP implementations (except from some life-cycle models), and no suggestions can be found on how it could be done in a better way. Hence, the aim of this thesis is to increase our understanding of how ERP implementations are engaged; how implementation teams actually make decisions about configuration, customization and organizational change. The scope being that a better understanding is necessary in order to develop new theory and better implementation approaches.

1.2 Research Question

The thesis' main research question is:

> How do organizations engage ERP implementations, and why does it often result in misfits?

Based on a case study and a focus group study this thesis provides new insights how organizations actually engage ERP implementations. But it also provides argumentation on why it is necessary to re-conceptualize ERP implementations as well as the objectives when implementing ERP-systems.

My empirical material shows that the perception of the ERP software as "standard software" providing "the best" way to perform a work process has significant implications for the way the consultants approach the cooperation with the user organization. First it results in a lecturing know-all attitude toward the user organization Second the cooperation with the organizational user representatives is organized in a way that privilege the ERP consultants needs for information to populate the configuration tables. This approach makes it very difficult for the organizational representatives to develop a thorough understanding of the socio-technical options provided through configuration (and customization), and they are given very difficult conditions for negotiating different options within the organization. As a result the realized mix of configuration, customization and organizational change is somewhat fortuitous and it is not committed in the organization.

If ERP software is regarded as “standard” software providing “best practice”, and implementing the software is understood to be a matter of re-engineering the organizational business processes, then some very important points are missed. ERP software is not “standard” in the sense that it provides a solution; it provides a repository of organizational work patterns that organizations can combine to become a solution, and the organization can decide on customizations if necessary. Second, implementing ERP-systems is not about re-engineering individual business processes, it is about creating an organizational wide process protocol that provide a basis for coordinating work in heterogeneous and distributed environments. Thus an ERP implementation is a design process with two interlinked design objects; an IT artifact and a federal process protocol. Both these design objects (artifact and protocol) require conscious design work. Articulation theory can be used as a way to understand how process protocols are formed, and it allows us to understand the IT artifact’s role in the articulation process. Thus the IT artifact provide (1) a technology to carry the bearing structure of the process protocol, and (2) support for actors understanding how to perform work processes in a way that allow the protocol to have a coordinating effect.

Thus the re-conceptualizing of ERP implementations provide a coherent way to understand the interplay between the organizational work processes and design of the IT artifact, and the connection between problems observed after go live and the design process.

1.3 The hermeneutic cycle and the structure of the thesis

My realization process has been truly hermeneutic which has influenced the elicitation of detailed research questions. Especially four levels in the realization process are important for the reader to be able to follow; the first level of realization is related to a case study, the second level is related to a focus group study, the third level is about my development of an ERP articulation process theory, and finally the fourth level of realization is provided when applying the theory for understanding ERP implementations as an articulation process re-interpreting the empirical data.

The overall research approach used is described in chapter 2, but below a very short introduction to the four realization levels and the detailed research questions for each level is provided. The introduction is given here in order to allow the reader to better understand the structure of the thesis.

1.4 Level one research question

The research project started with a case study in a medium size Danish company. From a project management perspective the ERP project was a success; the initial set up of the ERP system was done within time and budget. The case organization, however, had a very rough period after go-live. Thus this case provided an opportunity to investigate the nature of the problems organizations experience after go-live, and insights provided was used as a basis for investigating if the origin of these issues could be traced back to the implementation approach. The detailed research questions for the first learning cycle:

1a) What are the misfits experienced in the user organization after going live?
1b) How is knowledge to design the system obtained during the ERP experience?

Thus the first learning cycle provide an understanding of the misfits experienced in the user organization after go-live, and establishes an understanding of the relation between misfits experienced and the approach used implementing the ERP system.

1.5 Level two research question

The next level of realization is related to a focus group study. The insights developed as a result of the first learning cycle gave rise to many new and more detailed questions. Some of the most urgent questions were related to the way shared insights were developed during the initial design process (design prior to go-live). One thing that puzzled me was why the ERP experts were dominating the design process. Another puzzling thing was the non-use of traditional design artifacts providing insights about the domain of use; how could it be that knowledge about the use context was almost absent during the initial design process.

ERP professionals seems to have decisive influence on the approach used thus I decided to ask a group of ERP professionals to help me develop a better understanding why specific techniques and tools are used during the initial design process. The research questions for the focus group study are:

2a) How do ERP professionals perceive ERP implementations; what are they about?
2b) How do ERP professionals perceive the need for knowledge integration in order to design the ERP system?
2c) Which design artifacts do ERP professionals use and why?

The second learning cycle provide an understanding of how ERP professionals perceive ERP implementations, and it establishes an understanding of the relation between their understanding of ERP implementations and the way they approach the cooperation with the user organization. This understanding includes the ERP professionals' attitude toward the user organization and the techniques and tools used to mediate the cooperation.

1.6 Level three research question

The knowledge I developed during the first two learning cycles made me realize that a re-conceptualization of ERP implementations was needed. Articulation theory was used to develop a new theory for understanding ERP implementations. There are no new research questions for this third level.

1.7 Level four research question

My new theory is then applied to re-interpret my empirical data. Thus two new research questions were asked to the existing empirical material:

3a) How is the ERP articulation process approached in the case organization?
3b) How does the ERP professionals' perception of ERP implementations correspond to ERP articulation work?

The fourth learning cycle provides a more nuanced understanding of how ERP implementations are engaged. Thus using my theory as a lens the empirical material reveals four different perceptions of how ERP implementations are engaged as an ERP articulation process.

A graphical illustration of the four realization levels are depicted in Figure 1.1 below. The numbers in the Figure refer to the detailed research questions.

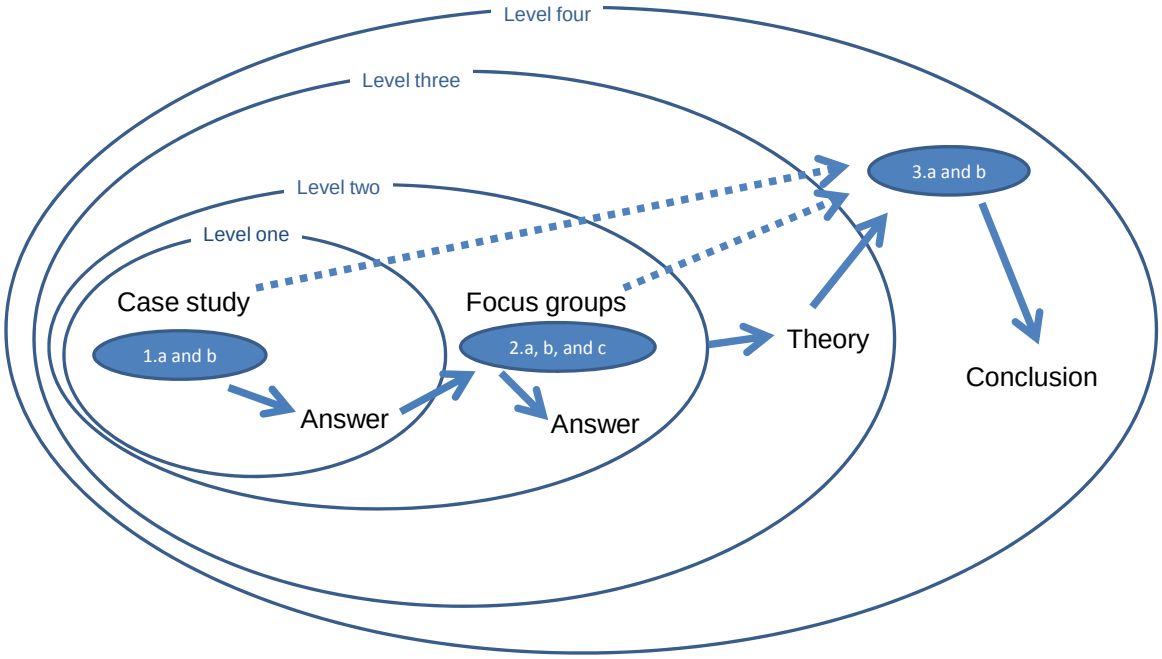


Figure 1.1: The four realization levels or learning cycles. Circles illustrate research questions. Full arrows illustrate the flow of the thesis. And dotted arrows indicate that the same data are reinterpreted.

1.8 Structure of the thesis

The thesis has six parts:

Part I - *Background and motivation* contains three chapters providing the basis for my research. The motivation for the research, the research goal, and the structure of the thesis is presented in chapter 1, the overall research approach and the research method for the first learning cycle in chapter 2, and an overview of the ERP literature in chapter 3.

Part II - *The first learning cycle* contains four chapters that together present the analysis of the case study. First an introduction to the case organization Alfa is given in chapter 4. In chapter 5 an analysis of the case data is performed in order to answer the first detailed research question (1.a). In chapter 6 another analysis is performed providing an answer to the second detailed research question (1.b). In this second part of the thesis I focus on providing an understanding of the misfits experienced in organizations after go-live, and establishing a relation between misfits experienced in the end user organization and the design approach used by the team engaging the ERP implementation team. In chapter 7 I conclude on the first learning cycle.

Part III - *The second learning cycle* contains three chapters. In Chapter 8 I discuss and present the research method for the second learning cycle. Then Chapter 9 provides the results from the focus group study; the answers to the three detailed research questions (2a, 2b and 2c). In this third part of the thesis I focus on the ERP professionals' (external and internal ERP experts) perception of ERP implementations, and how this perception influences their relation to the user organization and their preferences in tools and techniques used during the cooperative design process. Finally, in chapter 10 I conclude on the second learning cycle.

Part IV - *The third learning cycle about re-conceptualizing ERP implementations* contains four chapters. Again I start with a short discussion and presentation of the research method; that is chapter 11. Then in chapter 12 I present articulation theory as a theoretical framework and argue how ERP implementations can be seen as an articulation process. In chapter 13 I use the misfit categories developed in chapter 5 to show why the process perspective normally applied in articulation theory needs to be accompanied by a taxonomic perspective in order arrive at the necessary level of data "standardization" needed for coordination as well as decision support on all levels in the organization. In chapter 14 I finally present a theory for re-conceptualizing an ERP implementation as an *ERP articulation process*.

Part V - *The fourth learning cycle* contains three chapters. First – again – a short research method chapter (chapter 15) where I discuss how to apply the theory for my data. Then in chapter 16 I re-interpret the empirical findings provided by the case study focusing on how design artifacts are used to support the different layers and perspectives in an ERP articulation process. In chapter 17 I re-interpret the empirical findings provided by the focus group study focusing on how meta-work performed during the initial design process is perceived by the ERP professionals.

Part VI - contains the *Conclusion* in chapter 18. Here I conclude on the four learning cycles, give an answer to the overall research question, and sums up my theoretical and practical contribution. Finally, I have a short chapter 19 where I discuss the implication of my answer to the overall research question.

Chapter 2
Research method

This chapter presents the epistemological position underpinning this doctoral thesis and the research approach used carrying out the research. The epistemological position and methodological approach used is described in section 2.1. In section 2.2 I explain how the theoretical framework is linked to the empirical design. In section 2.3 the detailed research design for the first learning cycle is explained.

2.1 Epistemology and methodological approach

This thesis research is conducted using an interpretive research philosophy. The choice of research philosophy is influenced by my own epistemological and ontological stand. I believe that the social world isn't given; it can only be interpreted, and that the social world is produced and reinforced by humans through their actions and interaction.

As social processes can only be interpreted I need to get inside the world of those actors generating the social processes in order to develop an understanding. This means that I have to observe the actors in action, interview or otherwise get engaged with the actors. Thus as a researcher I will always become implicated in the phenomena being studied; my prior assumptions, beliefs, values and interests will influence my investigation. Hence I am part in creating the reality I am studying, and I can never fully retell the actors' story because my own interpretive schemes always intervene.

Within the interpretive tradition knowledge of reality is gained only through social construction such as language, shared meanings, documents and other artifacts, and the focus is on understanding the complexity of human sense making (Orlikowski and Baroundi 1991). Interpretive research aims to understand human thoughts and actions in the social and organizational context.

"The aim of all interpretive research is to understand how members of a social group through their participation in social processes, enact their particular realities and endow them with meaning, and to show how these meanings, beliefs and intentions of the members help to constitute their social action." (Gibbons 1987, p. 3 in (Orlikowski and Baroundi 1991)).

Interpretive research established itself as an important stand within information systems research in the 1990's ((Walsham 1995)), and is considered appropriate for developing deep insights into information systems as a phenomena.

"…it has the potential to produce deep insights into information systems phenomena including the management of information systems and information systems development." (Klien & Myers 1999 p. 67)

Field studies; in-depth case studies ((Walsham 1993)) and ethnographies (Suchman 1987) are one way to conduct interpretive research. Field studies are often associated with qualitative

research; and "qualitative" research and "interpretive" research is often assumed to be the same, but it is important to be aware that qualitative research can be conducted within a positivistic, interpretive or critical research paradigm ((Orlikowski and Baroundi 1991)). The criteria for evaluating research within these three different paradigms are very different, thus it is very important to be clear about which paradigm the research belongs to. As stated above, my research is conducted within the interpretive paradigm.

Interpretive research is often accused of being person-specific, artistic or private to a degree where it is impossible for others to judge the goodness of the results ((Miles and Huberman 1994)). Also within information systems research a call for explicit criteria for judging interpretive research has been put forward ((Lee 1991)). The glove was taken up by Klein and Myers who formulated a set of seven principles for conducting interpretive research of a hermeneutic nature focused on interpretive field studies ((Klein and Myers 1999)). The principles are meant as an inspiration to researchers conducting or evaluating interpretive research within IS. The seven principles are:

1. *The fundamental principle of the hermeneutic circle*
2. *The principle of contextualization*
3. *The principle of interaction between the researchers and the subjects*
4. *The principle of abstraction and generalization*
5. *The principle of dialogical reasoning*
6. *The principle of multiple interpretations*
7. *The principle of suspicion*

I have used these seven principles as an inspiration for conducting my research. In the next section I explain how I have these applied these principles in my thesis research approach.

2.2 How I used the seven principles for interpretive research

The 1st fundamental principle of the hermeneutic circle is applied to the research project as a whole as well as to the individual pieces of the research. Here I will explain how it relates to the overall research approach. For each of the four learning cycles a more detailed research design is presented in relation to each learning cycle (section 2.3, chapter 8, chapter 11 and chapter 15).

Klein & Myers (1999) suggest using the seven principles when trying to make sense of the empirical material. Each principle is intended to reveal different aspects of the field material. Thus each principle is intended to help the researcher better understand a significant part of the empirical material that contributes to the understanding of the field material as a whole. The way they suggest to use the framework is to apply principle 2-6 to be able to move back and forth between different interpretations of the empirical material resulting from applying the individual principles, and continue until the material makes sense to the researcher. I have tried to follow this advice; below I explain how I used the different principles in relation to the four learning cycles.

The findings provided by the first learning cycle draw my attention to assumptions and practices provided by the larger context; the rationale behind implementing ERP and the

generally recognized methods used to implement ERP. I suspected that the larger context could somehow influence what happened in the case organization, and decided to investigate **if** I could find support for my suspicion. Thus the research conducted during the second learning cycle (an overview of the cycles can be found in chapter 1; cf. figure 1) provides a contextual perspective on the findings from the first learning cycle (applying the *2nd principle of contextualization*), this then allowed another round in the hermeneutic circle to take place. The third learning cycle involved searching for theories that could provide a framework for better understanding the issues involved in ERP implementations and explain the difficulties arriving at a good socio-technical design. I ended up using articulation theory as a framework for generalizing my empirical material. Thus in the third learning cycle especially *the 5th principle of dialogical reasoning* and *the 4th principle of abstraction and generalization* is applied. Finally in the fourth learning cycle the new theoretical framework is used to show how the empirical data collected during learning cycle one and two can be re-interpreted providing a new (different) explanation to the overall research question.

The four learning cycles is one way to apply *the 5th principle of dialogical reasoning*, but this principle is also applied when analyzing the empirical data within a specific learning cycle. Thus as a researcher I strive to make the historical intellectual basis of the research visible to the reader and myself. E.g. when analyzing the data in the first learning cycle I started out interpreting the data using an existing framework of ERP misfits, but quickly I realized that the framework didn't fully explain my data. I examined my own point of departure and realized that I had applied a completely different basis for collecting the empirical data than the one that was embedded in the existing misfit framework. Therefore I needed to build a new framework appropriate for understanding misfits from my position. Thus I changed the approach and applied grounded theory building, and a new framework with four misfit categories was developed. Throughout the research project I strived to make my own prejudice and prior knowledge known to myself (and later to the reader).

The 3rd principle of interaction between the researcher and the subjects require a critical reflection on how data is socially constructed through the interaction between the researchers and the participants. I give an account of how this issue is approached when explaining the detailed research design for the first learning cycles in section 2.3 and for the second learning cycle in chapter 8.

The *6th principle of multiple interpretations* requires sensitivity to possible differences in the interpretations among the participants. This issue is addressed focusing on different actors in learning cycle one and two. I strive to uncover conflicting interpretations of the participants in the field. Thus in the first learning cycle I focus on different actors from the customer organization, and in the second learning cycle I focus on ERP experts. Especially during the second learning cycle I strive to examine the influence that the social context has upon the actors involved in ERP implementations.

The 7th principle of suspicion: This principle requires the researcher to be sensitive to potential "biases" and systematic "distortions" in the participants' stories. Klein and Myers (1999) leave it to the researcher to decide if they want to use this principle or not because there is disagreements among interpretive researchers conserving to which extend interpretive research can/should be critical. I have not applied a critical research methodology as such, it is more a matter of applying a critical perspective as suggested by Walsham (2005); a personal

motivation to focus on what is wrong e.g. issues of asymmetries of power, alienation, disadvantaged groups or structural inequity.

"In trying to carry out critical work, I choose a research focus but then allow it to shift over time, I try to remain open to what participants are saying, but I use theory to explore their perceptions and the context within which they are embedded. ….. all of the above would apply to "interpretive research" as well as "critical research". I would argue that what distinguishes the two, although they overlap strongly includes ……. researcher motivation, choice of research focus, theory selection, and active engagement with others to influence them, and ourselves, of the value of results from critical IS research" ((Walsham 2005) p. 116)

My research is characterized by an emphasis on the hermeneutic process, looking for a more profound meaning than that immediately given or conventionally understood. The empirical material is the subject of my attempt to assess meanings and develop insights, but critical elements have little importance during the main part of my research process. However, as my research project progressed it became more important to develop a deeper understanding of how and why the ERP experts are able to/allowed to dominate the design process. Thus a more critical perspective was introduced.

In Table 2.1 I have provided an overview of the thesis research philosophy and the overall research design. A detailed description of the research design for each of the four learning cycles are provided in the respectively parts of the thesis.

Research perspective	**Choice**
Topic	Implementing configurable software in organizations
Research question	How do organizations engage in ERP implementations and why does it often result in misfits?
Epistemology an methodological approach	Interpretive hermeneutic
Research design	**Four leaning cycles:** **First learning cycle** - field study: *Data collection:* Interviews and review of written documentation. *Data analysis:* • Research question one - grounded theory • Research question two - process model and knowledge framework **Second learning cycle** - focus groups: *Data collection:* Focus group discussions (with pre-defined questions) and exercises *Data analysis:*

	• Research question one: Mind maps (cognitive maps?) • Research question two: Mind maps • Research question three: Category evaluation; Analysis of the arguments behind the evaluation **Third Learning cycle** – theory development • Findings from the first learning cycle used to develop a framework understanding ERP implementations as an ERP articulation process (expanding existing articulation theory) **Fourth learning cycle** – Re-interpreting data: • Research question one: re-interpreting the case data using the theory developed during the third learning cycle providing an explanation how the ERP implementation is approached and why it results in the experienced misfits. • Research question two: re-interpreting the focus group metaphors as different ways to engage meta-work in an ERP articulation process.
Focus level	ERP implementation team and end-users
Analytical focus	Socio-technical design and user participation
Detailed units of analysis	Design work done by organizational representatives and ERP experts
Theoretical grounding	Articulation theory

Table 2.1: Overview of the research perspective in this Ph.D. thesis.

2.3 Research design for first learning cycle

The immediate aim of the research is to understand how organizations engage ERP implementations seen from the perspective of the actors participating in deciding on the design of the ERP system (the mix of configuration, customization and organizational change).

An interpretive case study was chosen because I needed to have details and in-depth knowledge to answer my overall research question. It is often assumed that case studies cannot be generalized, however generalizing form a case study to theory depend on the quality of the argument, not the number of observations (Baskerville and Lee 1999).

Following an implementation from the initiation until stabilized use would have been preferable, but since most ERP implementation take longer than a Ph.D. project some degree of pragmatism had to be applied when selecting a case organization. Thus I had to find an organization engaged in implementing an ERP system, who would also be willing to allow me to visit and interview actors participating in the implementation. By coincidence I got contact with Alfa (pseudonym), a Danish company who had started an ERP implementation a couple of years before my research project began, and luckily enough they allowed me to follow their project for almost three years. Although it was a bit of a coincidence I quickly saw that Alfa had the potential to serve as a good case; where 'good' means suited for answering my research question. Alfa was implementing a large ERP system provided by one of the key players in the ERP sector, and in a Danish perspective it was a relatively large and complex single site implementation lasting several years from project initiation to relatively stable use.

Relating to the hermeneutic concept of empathy my aim was to get close to the people and the organization implementing the ERP-system in order to understand the socio-technical design process from the perspective of the actors actively involved in the process. Thus the interviews had to include organizational representatives in the design team, IT-experts, and end users participating in the socio-technical design. Since so much existing ERP-literature (see introduction in chapter 1) have a management perspective I decided to delimit my selves from that. Thus project goals and business strategy given from management is only considered as design constraints.

Since the project started prior to the interview period one part of the interviews was conducted with a retrospective focus and another part focused on the current situation. One of the problems using this approach was that the interviewees' interpretations of the past were influenced by events taking place later. Written project documentation was therefore used to verify the interviews where possible, and contradictions and conflicting statements were put forward for the interviewees to comment on. Alfa provided elaborate documentation including detailed requirement specifications, documentation from the evaluation of the alternative candidate systems, business cases, gap analysis, and issue-log and change requests. All written documents were provided after the first interview, and were used preparing the following interviews. Thus the interviewee's role in the development of each artifact was disused with the interviewee, along with the perceived usefulness of the artifact in the design process, during training, and after going live.

The aim of my research was to gain a deeper understanding of the difficulties when implementing ERP systems. The related literature addressed the difficulties but did not provide explanations that could be used to improve the situation. I found that the literature on ERP implementations had a tendency to perceive ERP implementations as an organizational change project, assuming that the factual properties of the IT-artifact and the design process were of minor importance for the outcome of the implementation. Agreeing with Orlikowski and Iacono (2001), I focused on the IT artifact and the finalization of its design during the implementation process.

I decided to use a detailed interpretive case study in line with the interpretive tradition of information technology studies (Klein and Myers 1999). I focused on the participants'

descriptions of the implementation process and their expressed feelings and reflections regarding the usefulness of the tools and methods applied.

The study was carried out in the Danish Headquarter of an international engineering company. In January 2001, Alfa initiated the process of selecting and implementing a standard ERP system. In October 2003 they went live. In the following years, Alfa struggled to stabilize the system and improve the use and usability of the system.

2.3.1 Data collection

Data collection for the first leaning cycle was carried out through interviews with the ERP project manager, users serving as team leaders during the implementation (some of them later moved to an internal ERP competence centre), managers and end-users from all functional areas within the scope of the project, a consultant participating in the project on the vendor side, and the vendor's solution architect.

All together 19 interviews were conducted. All interviews were semi-structured and lasted 1½ to 2 hours. The interview guide included open-ended questions regarding experienced misfits, the interviewees' involvement in the ERP implementation over time, cooperation between user representatives and it-experts, tools and techniques used for requirements specification and design work. An excerpt from one interview guide is shown in Figure 2.1.

The Interviewee

- Name, function, years in company, participation in other projects, role in this ERP project?

ERP implementation

- To what degree have you been part of the ERP implementation process?
- How was it to be part of the implementation process?
- How was it to specify requirements for this project
- What did you do to make requirements (business) process oriented?
- Concrete example?

Context for ERP implementation

- How was the situation before implementation in CONTEXT
- Did you have a vision/scope for the implementation seen from your CONTEXT
- How is it after the implementation in your CONTEXT
- Was the vision / scope realized? Why / why not?

After (first) ERP implementation

- Have you had follow-up projects?
- Would you do something different today? In an ideal world?

.
.
.

Figure 2.1: Excerpt from semi-structured interview guide.

The interviews were taped, transcribed and verified by the interviewee. Table 2.2 shows the number of interviews within different groups and the timeframe.

Role in the ERP implementation	Number of interviews	Interview Periods
ERP Project Manager	3 interviews	February 2005 January 2006 May 2007
4 people from the internal ERP competence centre	(1)-3 interviews each	August - November 2005 June – August 2007
Vendors solution architect and a consultant	1-(2) interviews each	November 2005 February 2006 July 2006
7 people from the end user organization	1 interview each	February 2006 – June 2006
In Total	19 interviews each lasting 1,5 to 2 hours	

Table 2.2: Overview of interviews done.

2.3.2 My own background

Prior to starting my research project I worked as a practitioner in the area of ERP implementations for more than ten years. Thus I had to be very conscious about how my own pre-understanding of ERP implementations influenced the interview guide and the interaction with the interviewees during the interviews. When developing the interview guide and planning the interviews there is no doubt that I consciously as well as unconsciously used my experience to focus on issues that I understood to be problematic.

Thus in my pre-understanding a traditional requirements specification is very problematic to develop for the user organization, and furthermore I perceived the requirements specification (as a document) to be very difficult to use as a basis for regulating the cooperation between the client organization and the vendor organization conducting ERP implementations. Never the less the case organization had spend much time developing a requirements specification and in their own understanding (especially the project managers') it had been a real advantage for them because the vendor at the end had to pay for a large customization. Thus despite my somewhat negative experience with requirements specifications I was very curious to get to understand how the case organization developed the requirements specification and how they

used it during the implementation. Furthermore I was interested in understanding if the detailed requirements specification helped them develop a good understanding of the "fit" between the organizations needs and the capabilities of the ERP system.

It is also important to understand that I am brought up in a socio-technical tradition, and working with ERP implementations for a decade I had developed a somewhat critical attitude toward the implementation methods used. Thus I had a sincere wish to contribute knowledge to develop new approaches to ERP implementations that provided a better socio-technical design for end users.

Being aware of my own pre-understanding I strived to formulate interview questions that didn't give away my bias, and when formulating follow-up questions I considered carefully how to do it without judging what was already said. However, I understand that it is not possible to avoid influencing what data is being created. Simply by deciding on the questions I influence which aspects are highlighted and which are left out. Taping the interviews and transcribing them made it possible for me to minimize the consequences of the interpretation that takes place during the interview. Thus going back to the original transcripts several times during the research project, sometimes also listening to the original tapes it has been possible to interpret/re-interpret the original answers.

2.3.3 Data analysis

The data analysis took place as a hermeneutic interpretative process (Klein and Myers 1999); that is, the data analysis was an iterative process going back and forth between coding and collecting data. I adopted an inductive approach, and did not specify theory a priori to guide the data collection. As the data were analyzed, relevant theories were investigated.

I entered the research with a 'bias': being aware of the practical difficulties taking advantage of pre-defined ERP software, my intention was to understand how ERP systems could be implemented so that they were useful and easy to use for multiple groups of end users. As the analysis progressed, I consulted different streams of literature that could provide insight into the empirical observations, e.g., literature that considers ERP implementations, user participation, knowledge integration, and design politics.

The research started out with an aim to understand how the original (initial) properties of the IT-artifact could be investigated early in an implementation process to allow for an informed selection of the ERP package software and early specification of customizations/need for redesign of business processes. The research was expected to provide a list (categories) of misfits and their origin, which could be used to investigate where in the implementation process the misfits could/should have been discovered and addressed. Alfa had produced a thorough requirements specification and evaluation of candidate ERP packages, and kept records of change requests during the configuration process. Thus the interviews were intended to document misfits experienced by the organization after going live complimenting the written documentation and providing a more complete picture of the misfits. After the first 4-5 interviews it was clear to me that actors in the organization had very different understandings of whether something qualified as a misfit or not, and when asking into the origin of the misfits the implementation process rather than the initial factual properties of the

IT artifact seemed to be an issue. As a consequence my research focus changed somewhat and the first analysis focused on the end users and provided a list of design misfits experienced in the user organization after going live.

Realizing the socially constructed element built into the notion of 'misfits' the focus of the research changed from the artifact to the implementation process, and slowly a paradox emerged; Alfa had been very conscious about having users participate throughout the implementation process, the organization´s policy of design seemed to fit an approach having users participate not just to provide knowledge to professional designers, but also to influence the design. However the interviews left an impression of users becoming more and more frustrated as the process moved forward, and the initial analysis revealed that actions and events in the case were strongly influenced by prior actions and events in the implementation process. Especially the knowledge needed to decide on the design of the ERP systems seemed to be an issue. Thus I was looking for a way to analyze the data allowing me to focus on the implementation as a process and how knowledge was developed and integrated during the implementation process.

Hence the two different detailed research questions were formulated for the first learning cycle:

1a) What are the misfits experienced in the user organization after going live?
1b) How is knowledge to design the system obtained during the ERP experience?

2.3.4 Data analysis for the misfit perspective (research question 1a)

Analyzing the interviews for the first paper a data centric approach was used inspired by grounded theory.

A very early version of the misfit analysis is presented in the paper "*ERP misfits: What is it and how do they come about?*" (Pries-Heje 2006). The paper's underlying perspective was the inability of the IT artifact rather than the use situation, thus the categories of the misfits and their properties reflected this focus on the IT-artifact. As my research progressed it became clear that perceiving misfits from a use perspective instead of understanding them as factual defects of an IT-artifact provided a completely different lens; a lens that focused on ***the situations*** where the misfits were experienced and ***who*** was experiencing the misfit. As my aim was to understand how the misfits and the implementation process are intertwined the new lens provided much more useful results, as it implied different design aspects related to the use situation, and who and what might be involved in the design process. Changing the perspective and reinterpreting the data four different categories of misfits emerged. The basic issues found in the interviews were the same, but they are grouped differently and understood from a use perspective. Thus although the empirical material is the same the analysis included in this thesis has very few similarities with the initial paper.

The approach used had several iterations. First I read all the transcripts of the interviews carefully underlining (marked with yellow outliner) all parts of the text where a misfit was explained. Then I made a new document copying all the passages underlined. In the new document a table was created; each line contained a reference ID to the original interview, the

text underlined and a cell for a coding ID. Then I translated all the text in the new table to English (Table included as appendix A). As described above more hermeneutic circles were performed before useful categories emerged covering all misfits found. I did not use the interviewees own words as labels instead I tried to understand how and why a given instance in the eyes of the interviewee qualified as a misfit.

Slowly I realized that what seemed to be important was the use situation. Thus focusing on the use situation four different use categories emerged; individual users interaction with the software in order to perform the daily work, a specific functional group's use of the software to support their work practice, the software's support of coordination between functional groups, and finally an organizational wide definition and use of data (for daily work as well as decision support). I then again looked carefully at all the misfits one by one and assigned a misfit category ID to each of them. In some of the misfit categories also sub-groups were identified. When all the categories were specified then I checked that the categories were covering all the instances of misfits identified. The result of the analysis is presented in chapter 5.

2.3.5 Data analysis for the knowledge integration perspective (research question 1b)

As explained above the initial interpretation of the empirical material implied that knowledge issues played a major role in the implementation process, especially that the effect of a knowledge breakdown didn't necessarily show immediately. I therefore looked for a way to analyze the data that allowed me to focus on the process. Thus the description of the case is organized using a process perspective, and is inspired by the work of Markus and Tanis (2000). Their framework divides the ERP implementation process into four stages; "Project chartering", "The project" (configuration and rollout), "Shakedown", and "Onward and Upward". I chose to split "the project" stage into two episodes; the configuration and customization episode and the training episode. This split into two different episodes is important because a change in the knowledge integration issues and the participation was observed at this split. Thus I arrived at a process model with five episodes.

Within each episode, the analysis of the knowledge integration issues was based on Kensing and Munk-Madsens' framework (1993). The framework identifies six knowledge areas that should be covered and integrated when developing IT-systems. My perception of design as insight building fits well with Kensing and Munk-Madsens' understanding of IT-development as knowledge integration: bridging knowledge about the technology and the user organization as a base for the design of the future system. The framework provided a way to systematically investigate whether and how different tools and techniques are used to support the knowledge integration during the design process. The details of the framework are presented in the literature review in chapter 3, and the actual analysis of the empirical material in chapter 6.

In chapter 7 the results from the two analyses is used to provide a preliminary answer to the overall research question.

Then in chapter 8 I continue the discussion and presentation of my research method; but for the second learning cycle.

Chapter 3
What have been said before – A literature review

In this chapter first a review of the ERP research literature with specific relevance for my research project is provided. Then I elaborate on the difference between ERP implementations and traditional systems development and implementation. I also introduce some traditional IS research with relevance for my research.

The chapter is organized as follows. A general introduction to ERP package software and research around ERP systems is provided in section 3.1. In section 3.2 an overview of the factors found to influence the success of ERP in organizations, in section 3.3 the ERP lifecycle and findings concerning the ERP implementation process, and in section 3.4 an overview of the difficulties experienced in the organizations after going live. Then in section 3.5 my understanding of how ERP implementations are different/similar to traditional systems development is presented, and arguments that ERP research lack to provide a good understanding of the actual design process; how ERP experts and the participants in the organization actually cooperate to provide the knowledge deciding on the design.

3.1 What is an ERP-system

In the last twenty years, organizations throughout the world have implemented configurable software products that allow integration of major business processes across the organization, and provide real-time data sharing. This kind of software products is often referred to as Enterprise Resource Planning (ERP) system, although the software may have a wider or different scope than enterprise resource planning.

Rosemann (1999) defines ERP-systems as customizable, standard application software which includes integrated business solutions for core processes in an organization e.g. production planning and control, warehouse management and the organizations core administrative functions such as accounting, human resource management, etceteras. More synonyms for ERP systems can be found e.g. integrated standard software packages, Enterprise Systems, enterprise wide systems, enterprise business systems.

Seddon et al. (2003), however, emphasize the difference between software and an enterprise system, thus they define Enterprise Systems (ES) as large-scale organizational systems build around package software. The ES includes people, processes, and information technology. Many types of enterprise systems software can be found, e.g. Enterprise Resource Planning (ERP) software, Customer Relationship Management (CRM) software, and Electronic Patient Records.

Common for enterprise systems software are (Seddon, Shanks et al. 2003):

- A set of integrated functional application modules that can be used as an organizations primary engine for integrating data, processes and information technology internally as well as with external partners

- The software has extensive knowledge about business practice build into it; knowledge the vendors have accumulated from a large number of client implementations
- It is a semi-finished product the needs to be configured, customized and integrated with other computer-based information systems to met the client organization's needs.

3.2 ERP research

Both as a phenomenon and as a research area ERP is fairly new. However two comprehensive literature reviews have been published (Esteves and Pastor 2001; Moon 2007). Both reviews leave an impression of a very diverse and immature research area.

Esteves and Pastors' review is an annotated bibliography including papers from 11 key journals and 8 conferences within Information Systems. The review cover the period from 1997 to 2000, and a total of 189 papers are found. Only 21 papers are published in IS journals. In 1997 only a total of 5 papers were found but in 1999 it was up to a total of 89 papers. Thus ERP-research really picked up at the end of the 1990s. Especially the implementation phase attracted the researchers' attention (78 papers). More authors argue that a different approach from traditional systems development is applied (Davenport 1998; Gibson, Holland et al. 1999). Many papers are concerned with establishing an understanding of the critical success factors for ERP implementations (Holland, Light et al. 1999; Sumner 1999; Parr and Shang 2000; Parr and Shanks 2000). I will get back to both subject areas below.

The second comprehensive review by Moon (2007) includes research on ERP which has been published in any scientific journals (no conferences) independent of the research area, thus many journals outside Information Systems are included e.g. accounting, production management, operational research, and organizational research. The review cover the period from 2000 to May 2006, and a total of 313 papers were found. Moon identifies six themes in the research; Implementation, Using ERP, Extension, Value, Trends and perspectives, and Education. Moon (2007) finds that 40% of the research belongs to the implementation theme. ERP implementations are found to be the single biggest project an organization can launch; it requires a significant level of resource commitment and changes throughout the organization. Many stories about failed implementations including a few fatal disasters are told. Again "critical success factors are found to be a popular topic, but change management also seems to attract attention. A smaller stream of research is interested in ERP life cycle models.

When taking a closer look at the papers included in these two reviews it seems almost impossible to find any ERP research looking into the details about how the user organization and the ERP experts cooperate in order to develop knowledge to decide on the configuration of the system, and decide whether customizations are necessary; only general views and recommendations are provided. For example it is important to have business knowledge present (Shanks, Parr et al. 2000; Sumner 2003), sharing knowledge is important (Stefanou 1999), adopting the organizations processes to those implied by the ERP package is important (Markus and Tanis 2000; Parr and Shanks 2003). Kawalek and Wood-Harper (2002) is one of the few exceptions. They conducted a case study of a multi-site roll-out, and found that business managers and other staff members were invited into (SAP) educational workshops in order to identify and express issues connected with the ERP implementation. Having

organizational staff participate served the interests of the project manager in reporting local circumstances (issues); finding the thorns. Other studies imply different approaches (Huang, Newell et al. 2001; Pan, Newell et al. 2001) focusing more on building paradigmatic overlap between technology experts and organizational participants. Thus although the emerging stream of ERP have provided insights into ERP implementations and the critical success factors in these implementations it does not pay attention to how ERP experts and organizational actors actually cooperate and develop the necessary knowledge to make decisions about the mix of configuration, customization and organizational change.

3.3 Critical Success Factors when implementing ERP

Interest in researching the critical success factors of ERP implementations has been strong from the very beginning, but it has changed character over time. Early research was characterized by case studies and surveys providing a (short) list of critical factors (Holland and Light 2003). Then followed studies summarizing earlier findings in order to establish accumulated knowledge, and studies trying to identify the novel risks for ERP implementations (Sumner 2003). Sumner (2003) found nine risk factors unique to ERP:

1. Failure to re-design business processes
2. Failure to follow an enterprise-wide design which supports data integration
3. Insufficient training and re-skilling
4. Insufficient internal expertise
5. Lack of business analysts with business and technology knowledge
6. Lack of ability to recruit and retain qualified ERP system developers
7. Failure to adhere to standardized specifications which the software supports
8. Lack of integration
9. Inability to avoid technological bottlenecks.

Lately more comprehensive frameworks has been developed e.g. linking success factors to specific phases in the ERP lifecycle (Al-Mashari, Al-Mudimigh et al. 2003; Parr and Shanks 2003).

3.4 The ERP life-cycle and the ERP implementation process

Models of the implementation process and investigation of issues related to different stages in the model is another research area with many research contributions. One of the more referenced models is developed by a group of practitioners providing consulting related to the ERP system SAP (Bancroft, Seip et al. 1998). The model has five phases:

1. Focus
2. AS IS
3. TO BE
4. Construction and Testing
5. Actual Implementation

The focus phase actually constituted planning; deciding on the project organization and developing a high level project plan. In the *AS IS* phase current business processes are analyzed, the software is installed, business processes are mapped to the ERP functionality, and the project team receives training. The *TO BE* phase includes high-level design and detailed design (compares to a requirement specification), followed by configuration workshops where the system is used as an interactive prototype. Then in *the construction and testing* the software is configured, interfaces are build and tested, and finally the system is tested. *Actual implementation* involves building networks and installing desktops, and finally the end user training is performed.

Markus and Tanis (2000) have developed a different ERP implementation model with four phases: Chartering, The Project, Shake Down, and Onward and Upward. The model is depicted in Figure 3.1.

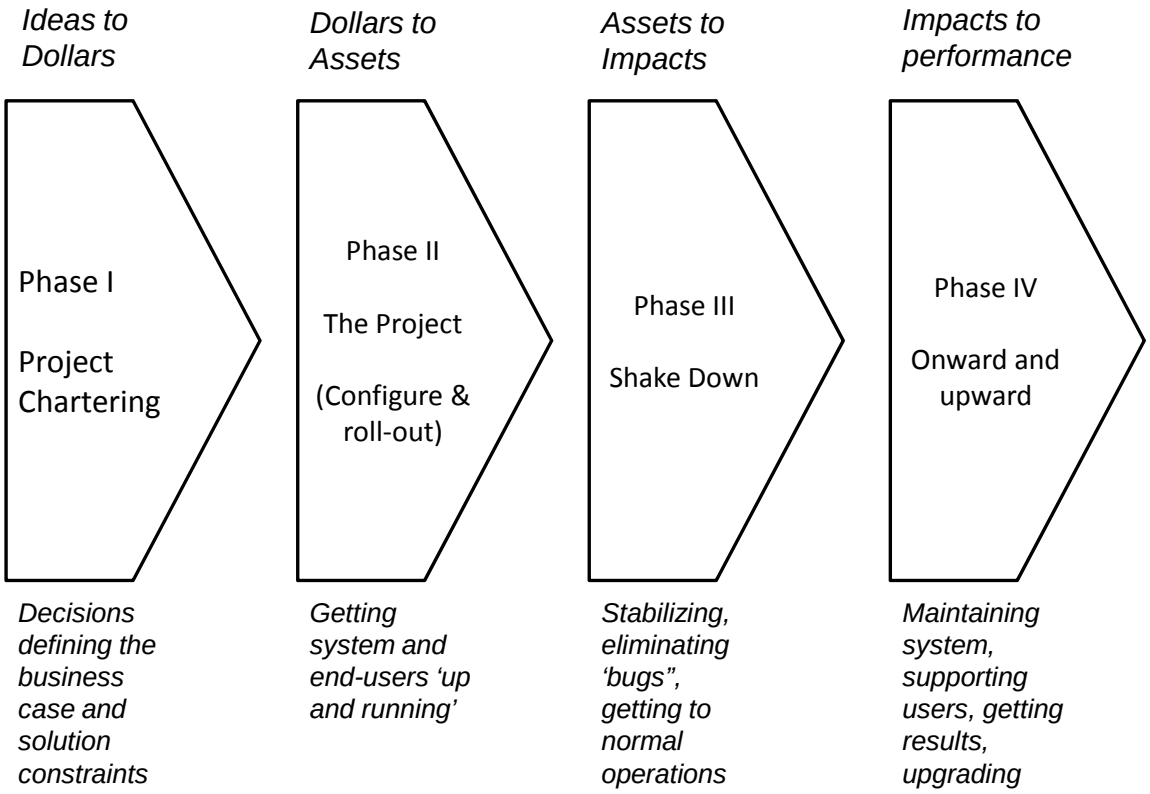


Figure 3.1: Model from Markus & Tanis (2000)

In a different study Markus et al. (2003) used the ERP implementation model investigating how successful companies are at different points in time in their ERP experience, and how different measures of success are related. They found that ERP implementations are enormously complex and nearly affect every aspect of organizational performance and functioning, and measuring success must reflect this fact. Thus different measures of success are defined for each phase. The study's conclusion is that outcome measured at one point in time is only loosely related to outcomes measured at another point in time. The reason for this is that the experience cycle is a set of processes, not a mechanical connection between starting conditions and final result, and during this process many (unforeseen) things can happen to influence the outcome.

3.5 Misfits experienced by the organization

Historically adopting Enterprise Systems has been associated with “misfits” although there is no common understanding of the term. Thus many companies experience moderate to severe business disruptions when going live with ERP package software, and they have difficulties recovering (Markus, Axline et al. 2003). A survey among 126 business managers made by (Keil and Tiwana 2005) shows; that managers considered functionality the most important attribute of standard software in order to predict perceived value of the system. However, the understanding of what constitutes functional fit or misfit and how to investigate the functional misfit throughout the life-cycle of an ERP system remains unclear. Some research has focused on misfits at the strategic level (Davenport 1998) and others at the operational level (Soh, Kien et al. 2000; Kien and Soh 2003).

3.5.1 Strategic fit

One research stream has applied a managerial perspective and focused on the strategic fit. Thus when implementing ERP systems one of the key issues for the organizations executive leaders are found to be whether the investment will pay off (Markus and Tanis 2000), and central to achieving the expected benefit is the development of a proper fit between the organizational strategy and the ERP system (Somers, Nelson et al. 2003). To realise the strategic fit and the expected benefits Somers and Nelson´s (2003) ERP fit model suggests a number of integration mechanisms to ensure the fit; Business driven implementation, project organization, package adaption and organizational adaption. Davenport (1998) on the other hand focuses on the characteristics of the software (at the outset) and its ability to fit the strategy. He argues that the system imposes its own logic on the company’s strategy and may push the company in an unwanted direction. Focusing on the benefits (mainly return on investment) Shang and Seddon (2002) propose a framework for enterprise benefit including five dimensions: Operational benefits, Managerial benefits, Strategic benefits, IT infrastructure benefits and organizational benefits. The framework was used in a survey including 233 organisations who implemented an ERP system. All organizations claim to have achieved benefits within at least two categories. Operational and infrastructure benefits were the most quoted benefits (73% and 83%), and all five categories were represented. There seems, however, to be a difference in the kind of benefit an organization is realising depending on which system they implement, thus indicating that the choice of system matter.

3.5.2 Functional perspective

Another stream of research has used a functional perspective on misfits. Several studies have investigated the perceived impotents of functional misfits (Bernoider and Kock 2000; Chang, Gable et al. 2000; Keil and Tiwana 2005). Common for these surveys are that they all confirm that functional misfits are perceived as a major concern among the respondents, but they treat misfits as a black box, leaving it to the implementing organizations to figure out what categories of functional misfits to look out for and how to do it. Soh et al. (2000) and Kien & Soh (2003) are some of the few who have tried to clarify the source and nature of misfit. They

use a cultural (national) perspective. Three Chinese hospitals implementing the same ERP system is used as a case to show how misfits may arise from differences in the culture (geographical location) that the ERP-system is developed in, and the culture (geographical area) it is implemented in. Kien and Soh (2003) found that misfits are largely due to differences between the implementing organizations context and the context the ERP package software was developed for. They identified four categories; differences related to nationality, sector, industry, and organization specific preferences. Misfits arising from differences in country, sector or industry seems to be more pervasive than idiosyncratic misfits reflecting differences in strategies, management preferences or user composition. As a basis for identifying the misfits they used documents specifying the approved customizations in each hospital thus their study omit the customization request but not approved, and the misfits uncovered after going live.

A third perception of how to understand and address misfits can be found in the research stream focusing on business process re-engineering and change management. Bancroft (1998) consider re-engineering and organizational change to be absolutely essential to ERP implementation success: "*reengineering (and implementing R/3 is a form of reengineering) must be seen as a mechanism for organizational change, Teams doing so are more successful than those that do no*t" (Bancroft, Seip et al. 1998 p 128). This perspective is also reflected in much of the research on success factors where "business re-engineering" and "change management" is found very often (Somers and Nelson 2004; Ehie and Madsen 2005; Al-Mashari 2006). Indirectly they imply that resistance to change or lack of knowledge on how to perform a business process is a main cause of misfits.

Difficulties investigating functional misfits in a specific organisation involve deciding what qualifies as a misfit. Asking different actors in the organization often illustrate that what some actors considered a misfit others perceive as an example of resistance to change. It seems that the more distant an actor is from the actual operation the more likely difficulties in using the software are considered a matter of resistance to change (Suchman 1995). Another difficulty arise from stakeholders having conflicting requirements, thus resolutions have to be negotiated and tradeoffs may result in some users experiencing difficulties using the software. Allen (2006) in a case study found that three value conflicts could be found between functional areas; conflicts over work priorities, conflicts over dependency on the commitments of others, and conflicts over evaluation of fairness. When value conflicts were perceived to be in balance and legitimate the users chose to use the information system, while they rejected it if they found conflicts to be too intense.

3.6 Comparing ERP-systems and traditional systems implementation

It is often argued that ERP implementations are different from traditional systems development. Especially because it is perceived to be desirable or necessary to adapt the organizational processes to the ERP package software (Bancroft, Seip et al. 1998; Markus and Tanis 2000; Kawalek and Wood-Harper 2002; Parr and Shanks 2003; Fenema, Koppius et al. 2007).

However, ERP projects are in general expected to involve a mix of organizational changes, configuration of the ERP software and customizations of the ERP software. The overall result

of the implementation is to change the way the organization work, and provide a new information system that supports the new way of working. Deciding on alternative ways to realize the mix of organizational change, configuration of the ERP software and customizing the software could be perceived as ***design***.

Using a notion of design as *"the reformation of conditions for human (working) life"* (Bertelsen 2001 p. 16) ERP implementations would obviously qualify as design . This notion of design implies that design is about "*taking something from the use domain bringing it into the room or zone of design and change it, then bringing it back to use"*.

Constructing IT artifacts may in itself not qualify as "taking something from the use domain" and bring it into the room of design. But if focusing on the IT artifact (e.g. the ERP software) it can be considered a medium to realize changes in something from the use domain. Thus in the context of Information Systems development the design of an IT artifact can be understood "*as an activity oriented toward changing another activity through the construction and introduction of the new computer artifact"* (Bertelsen 2001 p. 16). Applying this way of thinking in the context of ERP implementations design could be understood as the activity oriented toward changing the organization's way of working through the construction (configuration and customization) and introduction of the ERP system.

If design as an activity is understood "*as a process where a designing subject shapes the design object by means of some design artifacts"* (Bertelsen 2000), then ERP implementations raise some interesting questions about the design subject, the design object and the design artifacts used to mediate the process.

If we take a starting point in the design object (the object formed in design) one may be wondering whether the *IT artifact* or the *organizational work processes* are the design object. It is often a presumption that the ERP package software contains pre-defined functionality that should/could be used by the implementing organization. It is also assumed that the IT artifact is already designed and what needs to be changed are the work processes; following this argument the design object is the organizational work processes. On the other hand the implemented version of the ERP system; the version the organization are using doesn't come out of the box, it has to be scoped, configured and populated with basic data (that is unique to the organization), and furthermore in most cases the software is also customized. Thus the functional properties of the ERP software used in a specific organization are unique to the organization; the basic ERP package software and the organizations' specific version of the ERP software are not the same. Implementing ERP systems both the organizational processes and the ERP software seem to be the object of design, and to some extent the two design objects have to inform each other. Which of the two design object has priority is unclear.

Focusing on work processes as the design object and using the notion of design as transformation then implementing ERP package software can been seen as a way to design/implement new work processes starting out from a repository of more or less generic work processes provided by the ERP software. It is presumed that ERP package software provides a shared database and integrated patterns of task organization for the majority of the work processes performed in an organization. The "work processes" provided by the ERP system are however no more than abstract patterns of task organization. Actual work processes as they are performed in organizations are situated; depending on the local context,

and is a result of decentralized negotiation (Gerson and Star 1986). Thus actual work processes may include ad hoc decisions by local actors, responding to local management preferences, organizational policies, rules and regulations and the limits of local information systems, and different actors may have different perceptions of the work processes. Making the whole complex of abstract work processes provided by the ERP package software operational in a specific organization it is necessary to customize (adapt) the processes to local circumstances.

As explained above when implementing ERP Package software in an organization mutual adaption of the ERP package software to the organization and the organization to the ERP package is anticipated implying a duality in the design object.

3.7 User participation and knowledge integration in ERP implementations

Having organizational staff participate in ERP implementations are considered essential for success (Kawalek and Wood-Harper 2002), (Nah, Zuckweiler et al. 2003), (Robey, Ross et al. 2002) and is expected to provide a better fit of user requirements, achieving better system quality, use, and acceptance (Esteves-Sousa and Pastor-Collado 2000). The design team should be balanced or cross-functional, and comprise a mix of external consultants and internal staff; the internal staff should develop the necessary skills for design and implementations (Gibson, Holland et al. 1999), (Parr and Shanks 2000), (Sumner 1999). Both business and technical knowledge are important (Parr and Shanks 2000), (Sumner 1999). Sharing information among the various parties involved is vital and requires partnership trust (Stefanou 1999), and the team should be empowered to make quick decisions (Parr and Shanks 2000). ERP research on how to organize and support user participation in the context of ERP implementations is however very limited. Thus user participation and knowledge issues are widely recognized as important success factors, but very few insights how this is actually engaged in organizations are provided by the ERP research.

In more traditional systems development three arguments for user participation can be found:

1. A design argument
2. A political argument
3. A user acceptance (use) argument

The political argument has been put forward primary by Participatory Design (PD) research. PD is concerned with developing information technology "*with a more human, creative and effective relationship between those involved in technology's design and its use, and in that way between technology and the human activities that provides technological systems with their reason of being*"(Suchman 1993).

Three main issues have been addressed by PD (Kensing and Blomberg 1998):

1. The politics of design, addressing the interaction between power relationships in organization and society and technological development, and the empowerment of users as actors in these relationships in order to co-determine the technological development.

2. The nature of participation, addressing the conditions that different design constituencies pose on the cooperative process, and how cooperation between users and developers can be mediated in the different contexts.
3. Methods, tools and techniques, facilitating the cooperative design.

One of the central challenges of PD is the mediation of design cooperation between different professional practices. To develop a usable and useful product, expertise about the application domain, that is about the work practices of the use context and technical expertise have to be brought to bear on each other. Design artifacts have been developed that serve as boundary objects mediating cooperation across heterogeneous communities of practice. Both users and developers have to be able to contribute to the evolving software application, anticipate the implication of specific design decisions on the technical implementation and the changing work practice, and evaluate it with respect to these implications.

Within the field of Participatory Design (PD) issues related to the nature and reasons for user participation can be thought of in terms of three distinguished arenas (Gärtner and Wagner 1996):

(1) The **individual project arena** where specific systems are designed and new organizational forms are created
(2) The **company arena** where "breakdowns" or violations of agreements are diagnosed and hitherto stable patterns of organizational functioning are questioned and redesigned
(3) The **national arena** where the general legal and political framework is negotiated which defines the relations between the various parties.

In the context of ERP implementations all three levels may be relevant. In the individual arena (the work situation level), technology is used as an instrument and communication media supporting local work, the current organization of work is often taken for granted, and user participation is aimed at improving the work situation. In the company arena the use of technology depend on how different activities are coordinated and integrated in the local organization. Conflicting interests between stakeholders is not only playing out during development, but may also articulate themselves in the discussion of the overall organizational goals, which in many cases guide the selection of the technology and the local design (Bjerkness and Bratteteig 1995).

When implementing ERP systems the scope of the system is often the organization as a whole, thus the ERP system can be seen as a common system serving many heterogeneous user groups at the same time. When considering design politics and user participation the totality of the system could be addressed using a management perspective or it could be emphasized that there are several differing perspectives depending on various stakeholders' organizational positions and roles (Bjerkness and Bratteteig 1995). Using the later perspective the realization of the system would be a compromise between interests and needs of many different user groups, and the goal would be to balance these interests. This perspective is similar to the socio-technical approach which takes as a premise that employers and employees have a common interest in developing useful computer systems (Bjørn-Andersen and Hedberg 1977), (Mumford 2003), (Markus 1983). The socio-technical approach also addresses the organization as a whole, and within socio-technical research techniques for

stakeholder participation in the organizational arena has been discussed and developed. Although PD and socio-technical design might disagree on the existence and nature of a labor-capital conflict, in practice it is difficult to see the difference between the two approaches (Bjerkness and Bratteteig 1995).

In technology development projects the arguments for why and how users should participate vary. At one end of the spectrum workers participate solely to provide (professional) designers with an understanding of the local work situation. The design work is initiated by management or design professionals, and is carried out by designers. Users have no or very limited influence on the design and they are only participating when their input is considered valuable to the designer. At the other end of the spectrum users participate not only because their knowledge is considered valuable, but also because their interests in the design outcome are acknowledged. Thus users participate in negotiating and deciding on how projects are negotiated and supported, and they participate in all phases of a project.

In a review of ten PD projects Clement and Van den Besselar (Clement and Van den Besselar 1993) outline five basic requirements for user participation (the first three reiterated from (Kensing and Blomberg 1998)): (1) access to relevant information, (2) the possibility for taking an independent position on the problems, and (3) participation in decision making, (4) the availability of appropriate participatory development methods and (5) room for alternative technical and/or organizational arrangements (Clement and Van den Besselar 1993). Over the years different tools and techniques for participatory design has been developed within PD research.

Using a different lens user participation and user involvement is an important factor in systems' success, e.g. studying the decision processes around system development and implementation (Ives and Olson 1984), (Robey and Farrow 1982) or when studying organizational change (Zmud and Cox 1979), (Baroudi, Olson et al. 1986).

User participation and user involvement has been defined as two distinct terms by Barki and Hartwick (1989) in relation to IT systems development. "*User participation*" refers to the behaviors and activities that users perform in the systems implementation process, and "*User involvemen*t" refers to a psychological state of the individual, and is defined as the importance and personal relevance of a system to a user. Barki and Hartwick (1989) found that the influence of user participation on system use is mediated by user involvement and attitudes concerning use. Following this Vidgen et al. (Vidgen, Wood-Haper et al. 1993) argue, that there is a socially constructed element to IS use quality that is culturally influenced and dynamic.

Thus in the context of IT development user involvement and user participation is understood to be important; playing a key role in success or failure, although there still are some contradictions in the findings. However, successful user participation is however not easy. Newman and Noble (1990) found that conflicts may arise from differences in perspective, or when users have insufficient influence or power to control the development and implementation process. Along the same line of thinking Robey and Farrow (1982) pointed out that participation without influence is unlikely to lead to success. Cavaye (1995) found that where tasks were unstructured and only described at the strategic level, then the urgency for user involvement increased, whereas Ives and Olson (1984) found that if a system is well

structured and well defined then it is not necessary to involve users for the purposes of system quality but perhaps for system acceptance. Finally, Noyes et al. (1996) highlighted the difficulties of deciding how to involve users, and when they should be involved.

3.8 User participation and knowledge issues

The role of human knowledge and skills involved in the ERP life cycle is an underlying theme in Markus and Tanis' (2000) stage model of ERP implementations (see Figure 3.1). Without discussing specific methods, Markus and Tanis emphasize the challenge that the configuration of the standard package to the specific use situation requires mapping the organizational requirements to the systems' business processes and the terminology used by the vendor.

User participation and knowledge issues are also central to Robey et al's (2002) multi case interview study, including both successful and less successful implementations of ERP packages. Robey et al. (2002) found that all participating users had difficulties obtaining sufficient knowledge to configure the system and assimilate the new business processes and management structures. They argue that the presence of domain knowledge and successful communication between IT-experts and users are necessary to overcome *the configuration knowledge barrier* and, in turn, participation and social bounding are essential for successful communication. In their study the more successful companies had a large core team with diverse expertise, the team members were rewarded to stay on the project until the end and the team was staffed with respected business and technology managers. Robey et al.'s (2002) study emphasized the knowledge transfer from IT experts to user representatives for the configuration, and in order to overcome the so-called "*assimilation knowledge barriers"* an incremental approach as well as formal training should be used. Although Robey et al. (2002) claim that successful communication is essential, their study provides no understanding of how the user representatives and consultants actually develop means for communication or how knowledge is generated and integrated over time.

In a study by Pan et al. (2001) knowledge integration in ERP implementations is identified as a key problem. They found that knowledge is embedded in complex organizational processes, in legacy systems, in externally based processes, and in the ERP system. Understanding and sharing this embedded knowledge is important in order to integrate knowledge. Bringing key participants together and solving conflicts between the different parties involved is difficult but necessary. Based on their case study, they argue that relationship building is critical: inter personal relations (one-on-one) as well as community relations (group-based).

Huang and Newell (2003) studied knowledge integration processes within cross-functional projects including ERP implementations. They found that knowledge integration is essentially about engaging participants through the promotion of project benefits and management of social networks. Their research had a focus on the organizational members and the development of a shared understanding within the organization; thus, the nature and the design of the IT-artifact were not considered.

In a different case study Huang et al. (Huang, Newell et al. 2001) identified the main processes involved in cross-functional knowledge integration as: 1) the penetration of different boundaries to obtain required knowledge and support; 2) the expansion of different

paradigms to achieve shared understanding; and 3) the reconfiguration of organizational memory to create new organizational routines and knowledge. In their study, the knowledge integration between organizational team members and technology experts is addressed. The case indicates that the way in which the IT system was developed and modified caused difficulties between the technology experts and the users involved. Although the modifications were deliberated by users and technological experts together, it was often difficult for both parts to explain why these modifications had been made. Externalizing knowledge that had been collectively constructed was difficult because of the limited overlaps of background knowledge. Most of the modifications had not been documented. Thus, the retrieval of related information and the change management became very problematic. The paper (Huang, Newell et al. 2001) article stays on an abstract level; it provides no insight into how the modifications were designed and decided upon, nor does it specify which tools and techniques the team used to support this process.

Although user participation is widely recognized as a critical success factor, the literature provides very limited insight into how user participation can be organized and supported within the context of ERP implementations. Thus, the problems of ERP implementation have not been addressed as problems of multi-disciplinary design processes but as problems of involvement, change management and commitment.

	Users present work	**New system**	**Technological options**
Abstract knowledge	(2) Relevant structures on users present work	(5) Vision and design proposals	(4) Overview of technological options
Concrete experience	(1) Concert experience with users present work	(6) Concrete experience with the new system	(5) Concrete experience with technological options

Figure 3.2: Six knowledge areas from Kensing & Munk-Madsen (1993)

How to facilitate the cooperation between domain experts and IT professionals when designing computerized support for specific work places is one of the main research themes in PD. Kensing and Munk-Madsen (1993) formulate this challenge as a knowledge integrations challenge. They define design as *"bridge-building, since something new is created from two*

separate things" (1993) p. 79 . They claim that in the design of a new IT-system, three knowledge domains are involved: the knowledge about users' present work, the technological possibilities, and knowledge about the future system that the process results in. Knowledge generation depends on successful communication and *"successful communication depends on the ability to establish situations in which mutual perturbation trigger changes in the stats of those involved, which in turn lead to structural congruence among communicating partners"*(Kensing and Munk-Madsen 1993), p. 79. Therefore, methods, tools and notations supporting knowledge generation and integration are especially important. Kensing and Munk-Madsen distinguish between abstract and concrete knowledge in the three different knowledge areas. The six resulting knowledge areas (two-by-three) are shown in figure 3.2.

Concrete experience with the users' present work has to be acquired by the developers involved in order to be able to develop representations of this work relevant for the design process. Developers also need the knowledge to be able to understand the limitation that the work context poses for the technological support. The methods to achieve this learning are for example: Apprenticeship with users, participatory observation, and interviews (Kensing and Munk-Madsen 1993), p.81.

Relevant structures of users' present work address the abstract knowledge of users' present work. Kensing and Munk Madsen emphasize that the kind of structures that is relevant depends on the purpose of the software. The professional abstractions the domain experts use might be adjusted and complemented to represent structures of the use context that are necessary in order to design adequate technological support. The standard software engineering analysis notations and representations, such as dataflow and data models, or object and sequence diagrams, as well as less formalized representations like wall graphs and rich pictures (1993), p.81, can be used. In the context of ERP system implementation, models of the "as-is" business processes can be regarded as supporting the development and representation of this category of knowledge.

Concrete experience with technological options belongs to the professional realm of the developers. Users have to acquire knowledge in this domain in order to be able to anticipate the deployment of new technology to support their current work practice.

Overview over technical options supports the informed decision between different implementation alternatives. Kensing and Munk-Madsen propose literature studies as a way to address this area of knowledge. The technical options in the context of an ERP system implementation are constrained by the configuration and customization possibilities a standard package is providing. Reference models packaged together with the specific ERP systems can be seen as an attempt to show the different possibilities that a specific ERP system can support.

Visions and design proposals denote the abstract knowledge of the future software and its usage. Kensing and Munk-Madsen focus on representations that mediate cooperation in design. They suggest that software engineering design methods and notations should be complemented by prototypes; scenarios and system visions should be used as means to cooperatively develop the knowledge in this area.

Concrete experience with the new system will make visible how the work practices are influenced. Here mock-ups and also experience with prototypes and similar techniques allow users to evaluate the usage of the software under development with respect to their former and future work practice.

Configuration and customization of standard systems is not a well-researched topic in PD. The configuration and customization of ERP systems integrating business processes across different departments have to take a heterogeneous user community into account. Few methods have been developed and tested for this purpose. An exception is the acquisition and implementation of a Hospital Information System documented in (Krabbel, Wetzel et al. 1996; Krabbel and Wetzel 1998). Krabbel et al. (1996) report on the acquisition and implementation process. They propose combining observation and interview-based task analysis resulting in scenarios about the present work practice and cross professional workshops. Moderation in the common workshops has to focus especially on allowing different members of the organization to contribute, thereby allowing them to put forward requirements and constraints in relation to their specific work practice. Krabbel et al. propose what they call 'point of view' pictures as a means of representing professional and role specific perspectives. The cooperation between different user groups and the change in their cooperation through the introduction of the standard system needs to be represented as well. Here, Krabbel et al. propose using a specific adaptation of rich pictures to visualize different channels of communication around complex tasks – like admitting a patient to the hospital – and the changes that will be implied through the implementation of an information system.

The follow-up article from 1998 (Krabbel and Wetzel) indicates that the task analysis did not prevent the implementation process from becoming problematic. The authors mention, for example, the lack of adequate specification of customization tasks (in their definition comprising both customization and configuration), problems with the flexibility – or rather the inflexibility – that the software provides, and organizational change.

Although the implementation of ERP systems is not well researched in the PD community, PD provides a number of concepts, methods and tools to facilitate the cooperation between IT professionals and domain experts, which can provide inspiration for the facilitation of ERP system implementation understood as design. In the analysis of case study, I use the framework developed by Kensing and Munk Madsen (1993) to show that problems experienced during the ERP implementation can be explained as design issues related to knowledge development and knowledge integration problems.

Part II - The first learning cycle

Part II - *The first learning cycle* contains four chapters that together present the analysis of the case study. First an introduction to the case organization Alfa is given in chapter 4. In chapter 5 an analysis of the case data is performed in order to answer the first detailed research question:

(1a.) What are the misfits experienced in the user organization after go-live?

In chapter 6 another analysis is performed providing an answer to the second detailed research question:

(1.b) How is knowledge to design the system obtained during the ERP experience?

Finally in chapter 7 I conclude on the first learning cycle.

These two research questions – 1.a and 1.b – are carefully framed to support each other in establishing a relation between misfits experienced in the end-user organization and the design approach used by the team engaged in the ERP implementation. Thus the first question provide a base for understanding what kind of design issues the organization face after go-live, and the second question provide a base for understanding the design process that led to this kind of design issues.

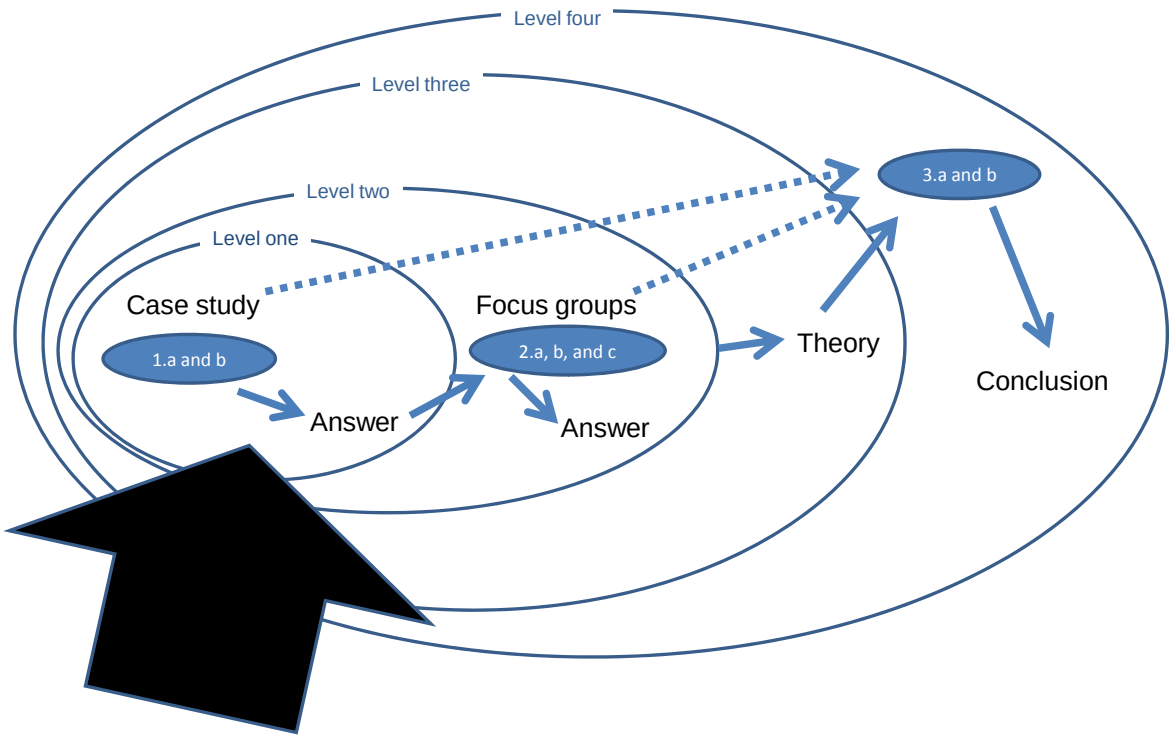


Figure II.1: The overall structure of the Ph.D. thesis with a thick arrow indicating where part II belongs

Chapter 4
Alfa – An ERP implementation case

In this chapter the case organization and the ERP implementation process used in the case organization is introduced. In section 4.1 a general introduction to the case company, then in section 4.2 the ERP project organization and the ERP lifecycle is presented, and in section 4.3 to 4.7 each of the episodes in the lifecycle is described in more details.

4.1 Alfa – The company

Alfa is an engineering company with more than 80 years of experience in supplying engineering services to the pharmaceutical and biotechnological industry. The organization has 1200 employees in Europe, China and USA. A large number of the employees have a degree from typically a technical university. Most of the work in Alfa is conducted in large projects lasting several years and costing 100s of millions US$.

In 2000 it was decided that Alfa should implement an integrated ERP package providing real-time data sharing in order to enhance the quality of services offered to the customers, improve resource management, and provide better financial control. Managers as well as users were aware that implementing ERP software would require some adaptation by the organization to the ERP system but at the same time they were also aware that the organization might have some unusual characteristics that they wanted/had to preserve.

The ERP project started in January 2001 at Alfa's headquarter in Denmark. A project manager with extensive ERP project management experience was hired. From the very beginning, the ERP selection and implementation were regarded as a joint project for management and employees in Alfa. It was never questioned that users would participate throughout the project as they always had in comparable projects. A project organization was set up and user representatives for each functional area were appointed.

Alfa's core business is project administration and project management on behalf of their customers. The company does not produce or manufacture any physical products. Alfa was aware that ERP systems in general don't have this line of business within their usual scope. Thus a thorough evaluation and selection process were conducted to ensure that the standard system would meet their needs. Alfa spent a year specifying requirements, evaluating candidate systems and selecting a system. It took more than 6 months to arrive at a final approval of the project, and then 9-10 months for configuration and customization of the system before going live in October 2003. In 2004, a follow up project was carried out addressing some major issues using the system, and since January 2005 the organization has continually been implementing (minor) re-designed functionality as well as new functionality.

Until the decision to implement the ERP system, Alfa had very limited experience with standard systems, and no experience with systems that works across functional areas in the organization. Only few user groups had concrete experience with modified standard software used in their daily work. Users, the ERP project manager and top management at Alfa

acknowledged the need for a new system and the intended approach; thus, the project started out being widely accepted.

4.2 The ERP implementation organization and ERP life-cycle

When Alfa decided to implement an ERP system, they also decided to have user participation in all phases of the implementation. Alfa's top management and the ERP project manager considered user participation essential for the quality of the solution as well as necessary for assimilation of the system in the organization. Thus during the project users participated in requirements specification, evaluation of candidate systems, scoping of the project, configuration of the system, testing of the system, and user training in the new system.

In figure 4.1 Alfa's ERP project organization is depicted. The steering committee consists of 4 persons from top management and the reference group of 12 influential and knowledgeable people from Alfa-s organization. Both the steering committee and the reference group used less than one hour a week on average throughout the ERP implementation. The ERP project manager and key people in the business group and the technical group are allocated 100% to the project, while a small number of people within each functional area are involved ad hoc.

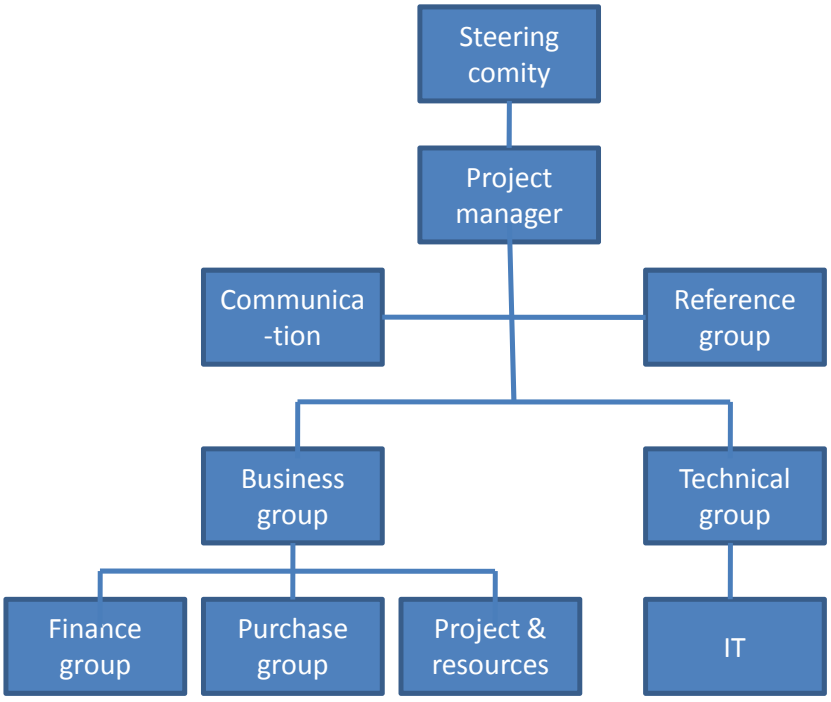


Figure 4.1: Alfa's ERP project organization

Using the ERP life-cycle model (Figure 4.1) Alfa's ERP implementation contain the phases shown in figure 4.2.

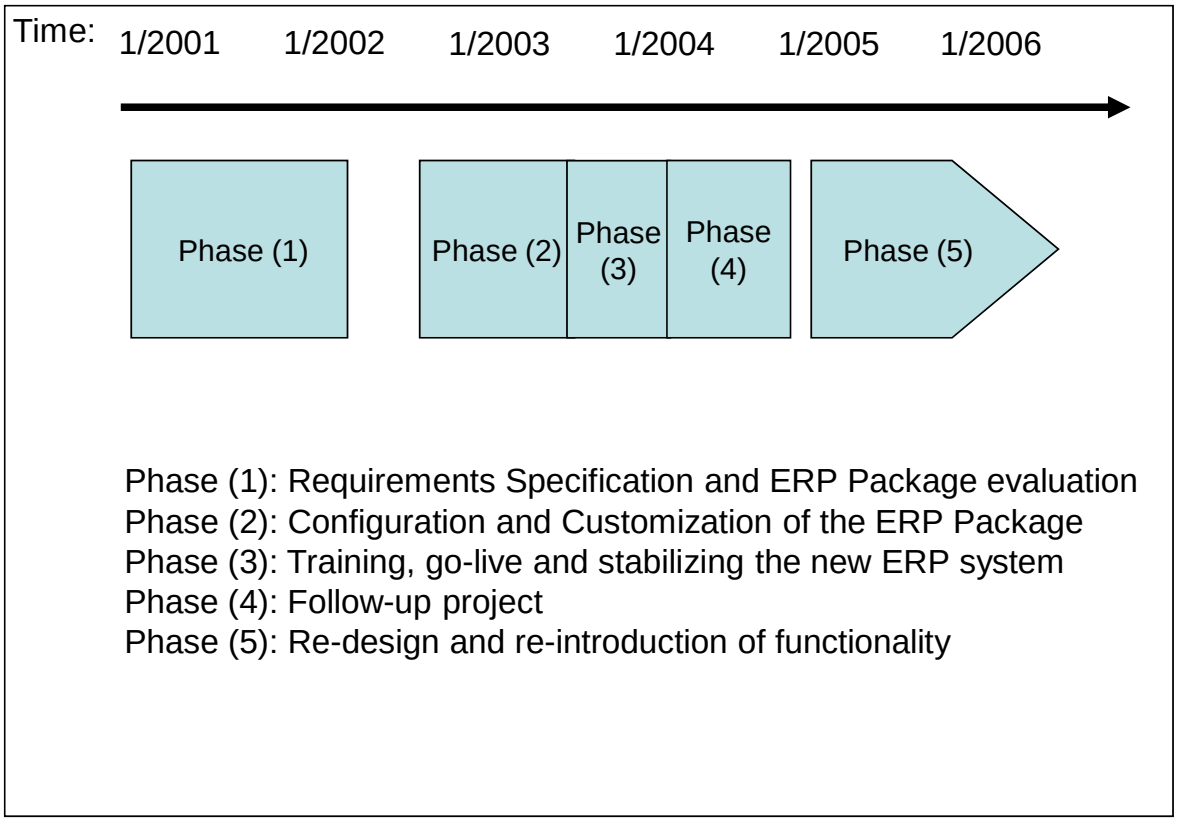


Figure 4.2: Time line for Alfa's ERP implementation

Below each of the phases in Alfa's ERP lifecycle is described in more detail.

4.3 Phase (1): Requirements specification and systems evaluation

First, all business processes within the scope of the new system were described using PowerPoint as a tool. The processes were related to four areas: Finance, purchasing, project administration, and resource management. A large number of users throughout the organization were involved in the process. For each of the four areas 'knock-out criteria' were defined. The business processes served as a common reference for discussing the requirements focusing on input (data) triggering a process, steps within a process and output from a process.

More detailed requirements for each area were defined in a dialogue between the project manager and the participating users. This turned out to be a difficult process as it involved a large number of users who had little or no experience at defining requirements. Alfa strived to have the requirements reflect existing processes and at the same time be forthcoming towards processes within a standard system. Because of the users' limited experience with integrated standard systems, they did not know what to expect from an ERP package. To inspire them, a few ERP packages were demonstrated by different vendors.

Alfa defined more than 800 detailed requirements that were subsequently prioritized on a scale from 1 to 4. The requirements were then mailed to the candidate vendors, and the

vendors sent a written reply; for each requirement they defined to what degree it could be fulfilled by their system.

In parallel with the requirements definition, a set of criteria for evaluating the vendor was defined. Knowledge about the industry and the vendor's desire to understand Alfa's business were among the more important criteria.

Based on the vendors written replies, three vendors were invited to demonstrate their system in an all-day workshop partly using material defined by Alfa. 10-15 users participated in the workshops and evaluated the system and the vendor's performance based on an evaluation framework. A group of three people (IT manager, project manager and a user representative) visited implementations of the candidate ERP systems.

The results from the evaluation process were summarized and presented as quantitative and qualitative scores in a number of different areas. A recommendation to the board of directors was subsequently made. The recommendation was almost unanimously only very few people talked in favor of a different vendor's product.

4.4 Phase (2): Configuration and customization of the ERP package

Alfa's board of directors decided to follow the recommendation given by the project group, and Oracle was chosen as Alfa's new ERP system. Some of the users participated in general training (3-5 days) in using the ERP system provided by Oracle during the time that the contract was being negotiated. Due to financial difficulties, the ERP project was asked to cut the project cost by about 700,000 Euro before starting. To re-scope the project, Alfa's ERP project manager and user representatives from the different functional areas together with ERP consultants conducted a 'Conference Room Pilot'; for each requirement, the implementation consultants would show their solution in Oracle. Possibilities of trimming the scope were discussed. This process made visible that it would be necessary to add some additional requirements as well. False assumptions of what would be provided by an ERP system had led to missing requirements. At the end of the two weeks, the revision was decided upon and a contract defining scope, price and so on was signed.

In the following nine months (in phase 2, year 2003, see figure 4.2), three more Conference Room Pilots were conducted. They can be seen as iterations in the configuration process. Each time, the system 'to-be' was (re-)scoped at a more detailed level and the configuration decisions were documented. The work was conducted in small workshops with user representatives and the consultant(s) from Oracle focusing on a specific module of the ERP package. As a part of the implementation method, the configuration and walkthrough of the system took its outset in the business processes pre-defined in the system. Oracle's process tool was used, linking the business process diagrams to the application. Alfa appointed one of the employees in the organization 'process integrator'. The process integrator had the task of focusing on the interfaces and coordination between processes and at the same time ensuring that all processes were signed off by a process owner. Alfa documented the new processes and the configuration decisions very conscientiously.

Teambuilding activities were conducted throughout phase 2. Although the project was under time pressure, the Oracle consultants and Alfa's user representatives worked together in a good atmosphere. The project manager worked explicitly with the aim of creating a team spirit.

4.5 Phase (3): Training and go-live

At this point the project was under extreme time pressure. Within Alfa, it is not allowed to implement a new financial system during the last quarter of a financial year. Therefore, the system had to go-live at the beginning of October 2003. The training of the users took place alongside the final testing and data conversion. On the 8th of October 2003, Alfa's Oracle solution went live. Because of the time pressure, many reports were not yet implemented and consequently much promising functionality was left to a later phase.

An important change in the ERP organization and the roles of participants happened during this episode. The external IT-specialists stepped back a little and allowed the user representatives to take over the role as ERP specialists in the organization. Thus, user representatives developed the training material and performed the training of end users. At the same time end-users entered the stage.

During training, a lot of resistance toward the system was built up. Users perceived the system's usefulness to be very low and prohibited at this late stage that the resource management functionality was taken into use. In their opinion, the functionality was too poor.

During the next yesr, the users were struggling with the system, learning to manage only some parts of the system. Other parts were rejected or used incorrectly which, in turn, caused data quality problems and malfunctioning in other areas. After a very turbulent period, the system was stabilized and the most important reports were developed. An internal 'IT-competence centre' was formed consisting of the project manager and some of the user representatives, plus a former Oracle consultant who was hired by Alfa. In general, the users had difficulties understanding how their personal use (or rejection) of the system influenced the work of other departments.

4.6 Phase (4): The follow-up project

The fit of the system was more problematic in some areas than in others. Members of the Oracle competence centre suggested to include all users in an evaluation of problematic issues regarding the system. Meetings were set up where people from the competence centre met all user groups within Alfa. The analysts met the users with an open mind. All issues reported were noted without considering the relevance or the reason. The process resulted in a list of more than 500 issues. Afterwards, the root causes for the issues were discussed and appropriate actions decided on. Some issues were obviated with end-user education; some with reconfiguration or customizations of the system; some were researched thoroughly, but could not be solved due to the core architecture of the ERP package. At the end of year 2004, this process was completed, although the re-design of some of the problematic processes was still outstanding.

In general, the users' perception of the usefulness of the software is slowly starting to change. However, many users still avoid using the system or enact it in ways that cause as little change to the old work processes as possible.

4.7 Phase (5): Re-design, re-introduction and design of functionality

In this phase the focus was on improving the use of the system; re-design of the already implemented functionality and design of new functionality. End-users throughout the organization and members of the competence centre were continually working to increase the quality of use. They customized the software to change the original capabilities of the ERP Package, reconfigured the system and used the software in unanticipated ways.

The relations between the users and the internal Oracle competence centre were still somewhat tense. The IT experts however considered the functionality within the new system to be useful for most parts. Users' complaints were seen as resistance to change; yet, some of the IT experts engaged in a dialogue with the users. Members of the competence centre communicated directly with the functional managers and end users to help them understand the new ERP software and the ERP package's capabilities. IT-experts observed users using the system and engaged in discussions. Super-users and end-users began to help each other on an ad-hoc basis across functional departments. Some users started to form groups to share their experiences. Employees from the competence centre and some end-users also participated in groups outside Alfa sharing experiences about the Oracle ERP system.

The general perception of the usefulness of the system improved although some users still performed their work in ways that minimized the interaction with the new system as much as possible, e.g., by asking their secretaries to fill in the data.

The users articulated that large parts of the new software and the derived work processes were poorly designed and a waste of their time. Together, the users and the IT-experts improved their knowledge about the use situation, the ERP package software and possible new socio-technical design possibilities. Some of them engaged in a dialogue discussing design possibilities and integrating knowledge from all three domains. The process was constrained by the lack of useful abstract knowledge about the ERP Package capabilities and the lack of abstract representations of the integrated work processes. Participation started to take place "by doing" (Ives and Olson 1984). As users also had to pay for the new development through their local budget, they gained strong control over the design and development process.

Chapter 5
A user perspective on misfits

The aim of this chapter is to lay out the *misfits* experienced in the user organization after going live and stabilizing the ERP system. The definition of misfits will be explained in more detail later, but for now it will be sufficient to think of misfits as a situation where the system's design are interfering with expedient work processes in the user organization. Given the wide spread notion of misfits and the limited research trying to understand what constitutes misfits in the context of ERP systems this chapter's result are valuable in its own right, but a better understanding of the misfits experienced in user organizations also provide a very important point of reference for the remainder of this thesis. E.g. when considering how the design issues experienced is related to the way the ERP implementation is approached. Thus the misfit categories identified provide allow comparing and contrasting different classes of misfits and consider how they come about, and make it possible to consider ways to overcome/approach the design issues during the implementation process.

The research question this chapter answer is:

> What are the misfits experienced in the user organization after go-live?

The method used to analyze the empirical data and derive the misfit categories was described in more detail in chapter 2. The documentation of the analysis can be found in Appendix A. All misfits experienced after going live are include, also those already solved (through a customization) at the time of my interviews. Both interviews and an "issue-list" developed by the organization after going live are used to provide the misfits.

As misfits may have a socially constructed element it may be difficult (impossible) to have all stakeholders agree in whether something qualify as a misfit. I have focused on the end-user organization, and if something in their perception qualifies as a misfit I have included it. The reason for including all misfits is that I am interested in understanding the difficulties experienced in the user organization, and if an actor perceive something as a misfit it indicate that there is an issue that needs to be considered.

The four categories of misfits identified are:

1. Misfits related to human computer interaction – individual level
2. Misfits related to a specific profession – group level
3. Misfits related to coordination between professions – inter group level
4. Misfits related to a shared taxonomy – organizational level

The remainder of this chapter is structured as follows. In section 5.1 – 5.4 the four categories of misfits identified are presented along with examples of each category, and in section 5.5 the findings are summarized and the research question is answered.

5.1 Category one: Design issues experienced by the individual

When focusing on the individual user three sub-categories of misfits are found. All three sub categories are concerned with the interplay of information technology and the human. The first two are directly related to the human computer interface provided by the ERP system while the last sub category has a broader perspective including all use of information technology and the coherence between the different technologies and the individuals work tasks.

5.1.1 The first sub category: Waste of time

This sub-group contain misfits where the users are complaining about time spend entering or finding relevant information; thus it is possible to perform the task but it takes much too much time.

Examples: The majority of Alfas customers are companies who need to build a large production plant to produce e.g. food, drugs or cement. The customer may not have the expertise to manage such a large project and therefore ask Alfa to do it for them. Alfa is not manufacturing anything for the project but they supply technical knowledge on how to build the plant and project management. A project may last several years and it is not uncommon for Alfa to have 200-300 people allocated to such a project at the same time. Thus allocating resources and updating the resource allocations are very important in order to get an overview of the project. This is however very cumbersome using the ERP system; the user have to use 4-5 different screens for each resource, for each task, and for each month. Prior to implementing the ERP system this work was done in a spreadsheet making it easy to use templates, factors such as '10%', and mass allocations. Thus spending several days every month updating the allocations in the new ERP system is considered an unnecessary waste of expensive time.

In addition to the large construction projects Alfa also conduct smaller maintenance projects and very small graphical projects. The users were expected to use the same screens for all type of projects, but that caused too much overhead for the smaller projects and new screens targeted had to be developed.

During the interview with the ERP project manager she explained that it has been necessary to make many customizations to the ERP system after going live because the ERP system originally had "*too many data fields and too many screens to navigate for the users"*. Also many new interfaces carrying over looked-up values have been developed. These kinds of customizations have been made using simple web-interfaces and an API.

5.1.2 The second sub category: dialogue incomprehensive to the user

This sub-group contains misfits where the human computer dialogue makes the users give up or introduce poor data quality. After going live several examples of this kind of misfit were experienced. In general the same screens (the user dialogue) are used for all users in the

organization. In most cases the super users are able to learn how to operate the ERP system and oversee an awkward interface, even learn to interpret error messages just stated as numbers. For occasional users however navigating the screens are difficult and it is very likely they make mistakes. When doing so they are meet by error messages that are targeted at an IT expert, thus often the occasional user has no clue what went wrong. Some occasional users give up where as others try over and over again introducing data quality problems.

Examples: The purchase order screen are the same for the users in the purchase department, and for the project manager buying materials for a large project and any user just buying something for internal consumption. The project managers are complaining about the usability and they are backed up by the super user in the purchase department who explains: "*The purchase order screens are difficult for the users: It is difficult to understand what to enter, how to understand error massages, choose the right item and/or vendor.*" And he continues "*This is causing poor data quality making it very difficult to use the data in the purchase department and the organization for decision support*".

Also the project controllers experience the problem. A controller explains: *"If people in the organization use the system correctly it is actually possible for us to generate a fair picture of the financial situation, but here the user friendliness kicks in e.g. around purchase. 50-60% of the project's costs are related to purchase and most users don't know how to do it correct."*

Project managers are also occasional users of financial data and other reports about the projects, and also here they experience difficulties. One of the controllers explains: "*Project managers use* (this part of – my addition) *the system very little and even if we* (the controllers – my addition) *come out again and again they just give up. The system requires a super user. If you are just using it from time to time you don't know how to use it and the system is not helping you."*

5.1.3 The third sub category: lack of cohesive force

This sub-group relates to obtaining a cohesive force between the ERP system, other IT systems and manual procedures. Although Alfa aim at having the ERP system support as many administrative tasks as possible more IT systems coexist, and in the eyes of the users this is sometimes causing difficult work conditions.

Example: Some of the workflows in the ERP system generate notifications that users have to respond to. To some of the users notifications seems as another e-mail application, and they resist having to work with two "e-mail" applications at the same time. The ERP project manager explains: *"They perceive it as an e-mail and they only want to read e-mail in one system! Thus we had to make a change tricking an e-mail when a notification was created."*

The project managers have difficulties gathering project data in order to provide a financial overview. "*In the old system you could get an overview over the financial status of a project. Now we have to look in 4-5 different places. ….. Finance is not their primary area therefore they need it to be simple*."

For the project controllers the work conditions are difficult: "*We use many Excel and Access tools. The disadvantage is that we spend much time moving data around. We use 70-80% of*

our time generating data; manually pulling data from all directions. This of cause includes a great risk that something is missed."

5.2 Category two: Design issues related to functional groups

This category contains misfits experienced by a specific functional (professional) group within Alfa, and relates to design issues making it difficult performing the work as a professional group in Alfa's specific context. As an example a functional group could be the people working in the purchasing department. Two sub-categories of misfits have been identified. The first sub-group relates to the tasks performed by the professionals; individual tasks or the sequence of tasks. The second sub-category relates to internal and external data presentation directly related to performing the tasks as a functional group.

5.2.1 The first sub-category: inadequate support for work tasks

This misfit sub-group contains design issues related to "what" work you do within a functional group and "how" you do it; the tasks performed and the sequence of tasks. Using the perspective of an individual functional group misfits in this category could result in tasks not recognized (supported) by the system's design, dispensable but mandatory tasks or redundant tasks performed.

Many examples of misfits in this category have to do with not being able to support scalability. An example could be related to projects performed in Alfa's organization, which can vary in size and complexity. The projects may follow the same sequence of tasks but the amount of data and the time allowed to perform each task may vary considerably. This issue is closely related to the human computer interface, but in this sub-category the focus is more on the detailed work performed during each task.

The basic design of the project functionality has caused difficulties for both smaller and larger projects. For the larger projects it has been decided to add ten screens providing data about the project; data the project manager need in order to cooperate and communicate with the customer. The work involved in the tasks related to the communication is however not supported by the ERP system. For the smaller projects cooperation and communication with the customer has another character and therefore these data is not needed. Thus the task entering the data is experienced as a dispensable task, but since it is mandatory they had to perform it (until another customization is made).

Some of the users tell about the difficulties with the project functionality:

"*Smaller projects have too much work and too little use of the data entered*".

"*One of the mini projects* (customizations made after going live) *was about an easier way to initiate a project* (for the smaller projects).

"*Changes happen fast; people are added to projects, new orders et cetera. Maintaining data is too difficult and time consuming. It is impossible to make cash-flow analysis."*

"Resource allocation is another example of a process that has difficulties scaling. You need to go through way to many steps before you have a correct allocation. Therefore a customization was needed to make it simpler (for the smaller projects)**.**

"*To begin with we had to enter the budget and periodize it monthly. If you have 400 tasks for each period and a project has 18 months duration, then you have 18x400 fields to update every month."*

Alfa's relation to its largest customer is providing some unique requirements concerning project management, reporting and purchase. Thus when taking the responsibility for project management on behalf of a customer Alfa's employees have to oversee the design of the production plant and subsequently find contractors who can deliver the parts needed to build it. Thus they negotiate a contract, oversee the delivery, approve the invoice and keep record over the costs. The invoice itself however should be paid directly by Alfa's customer.

Example: The original ERP software is unable to provide a solution where all tasks around purchase of materials are supported, including data for project controlling, and at the same time keep Alfa's financial books unaffected of the customer's purchase. Thus a customization is made before going live and as a result Alfa has to operate with two sets of "financial books"; one collecting Alfa's financial data and another directing the financial implications of the purchase on behalf of the customers into a "black hole". This customization makes workarounds necessary when using some of the ERP software's original functionality. Another result is that redundant work around recording and managing invoices form suppliers are introduced.

"The two books means that all projects has to be entered twice; in the ordinary financial book and in the fictive book, and be linked to each others. Furthermore we have to use the specially developed reports shoving the Alfa view."

"I think it is a very common issue (scalability)*. You make the system work well in a specific situation. The ERP software has different ways to deal with problems, and some of them we use, but it is not possible for Alfa to take advantage of them all because of the two set of books."*

At Alfa the majority of the materials (items) purchased are unique to the individual building project which make the anticipated budgeting and purchase process in the ERP software very challenging for Alfa.

When looking at the purchasing management "life cycle"; from budgeting to payment then the ERP software implies knowing the item number from the start and using the same item number all the way. In Alfas' case, however, this is not easy. At the time of budgeting there is often only a weak idea about the "product" (item) and the contractor. The process of finding possible contractors could start long before the actual specification of the product. Using the logic of discrete items, service items or production items (based on a bill of materials) is not possible. In this case it is more a "buy to order" and that kind of logic is not easily provided in the interplay between the project module and the purchase module. In Alfa's case the solution

designed by the implementation team is causing many difficult for both project managers (project participants) and purchase employees.

The business logic build into the ERP software is challenging Alfa's way of working in several functional groups, especially the way the item concept is implemented is problematic. Alfa is both selling and buying unique service items and this is introducing many peculiar difficulties.

Example: A painting job is ordered and a price of 100.000 kr. is agreed on. Then an invoice of 10.000 kr. is received meaning that 1/10 of the work is done. By default the system suggest that 1 piece is delivered, thus if "1" is not manually overwritten by 0.1 then the order will be closed, the order cannot be include in any further work, and if a financial report is drawn then it will show a wrong result.

Another complication is that it is impossible to use the number of items or the amount agreed on as an indicator to close the "order", the only way to know for sure is to get a message from the project manager. The software does not allow/support this communication.

Example: In the purchase department they intended to match the reception of the goods with the order, and in the software it should be possible to do two ways, three ways, four ways and five ways matching. You however have to choose one as default and use this for all items. Because of the almost equal mix of physical items and service items choosing one way to do it would cause difficulties in almost half of the cases.

The work in the purchase department and the way it is performed is influenced by workarounds and conflicting requirements between different departments. A super user explains: "*Workaround the purchase flow mean, that request for quotation is handled manually outside the system. It is primarily caused by the project manager or project assistants wish to follow their requests. The request number is lost when an order is made. This workaround result in the request and the order to coexist if the request is not closed (manually) when the order is opened manually.*

The department managers (technical departments delivering resources to the projects) have a difficult work process when approving timecards from their employees:

"Time-card issues could not be resolved. Projects have numbers and names. In the system's screens where the manager has to approve a time-card only the number can be shown. And when a manager approves 25 employees' projects all they can see is 50+ different numbers. It is impossible to remember. If they had seen the names, they would have known if it is correct or not."

Furthermore, in the financial department they have difficulties providing a profit-lose statement.

"I don't understand why we have decided not to have the project dimension used in GL [=General Ledger, the main accounting record of a business which uses double-entry bookkeeping]. If we had done it, then we would have had the profit-lose statement directly. It is the biggest mistake we have made designing the system".

5.2.2 The second sub-category: Inappropriate data presentation

This sub-category contains misfits related to internal and external data presentation; both reports and other kind of summarized data used to provide overview for people belonging to a specific functional group and external papers meant for customers and other partners interacting with the functional group. In the users understanding the data is stored in the ERP system's database, but cannot be (are not) presented in a useful way.

"*Especially the projects converted into the ERP system are problematic because you have to look more than one place for data, but also just getting an overview. I don't know if it is the capacity or the willingness that make them give up (the project managers). What they need is one place where they can enter the project number and the date, and then a financial statement comes out.*"

"*The difficulty getting the right data out of the system and be able to present them in a useful way is frustrating. I will always have to use Excel when making reports*."

"Regarding resources we are now getting to a point where we can enter data about allocations, but we cannot get a report out showing the resources needed over time e.g. budget versus realized data. But we have made an Excel report based on macros that will do it now."

"*We also lack proper reports for the external customers. A report has been developed, but I don't give it six month. It has to be changed. The report is specified by our VP. It is possible to make manual invoices but they are not included in the report.*"

"When we have to provide a detailed specification of the invoice to the customer (project owner) then we often lack the necessary information. I think data is in the database but it is not specified on the invoice."

"*The ERP system's ability to support presentation of data is too poor, we need more graphical representations."*

Many examples are given from different users how this is already done or can be done with Excel or Cognos.

5.3 Category three: Design issues related to coordinating work between groups

This category contains misfits experienced by functional groups in relation to coordinating work between the groups. The misfits are related to work processes where some kind of coordination mechanisms are necessary to allow different actors (groups of actors) to perform their task(s) in a shared process (a series of tasks) where the tasks are split between them. Processes cutting across functional groups may be implemented in the ERP software's basic application layer or in workflows implemented on top of the application layer. The misfits are

experienced by the users in many different ways: as missing, dispensable or redundant tasks, inappropriate sequence of tasks, data shared or exchanged is inappropriate for coordinating work or performing work locally. For processes implemented as workflows the misfits could also be related to inappropriate authorization hierarchy or inappropriate data controls.

Some examples given by the users:

"*For workflow around purchase we need a project hierarchy in steed of a department hierarchy* (as provided by the ERP software)*, but for other purposes we need the department hierarchy*."

"We had a situation where a project with 100 persons was budgeted correct and then a department manager was able to add 3 men 100% and all our work was wasted. They only needed to contribute a few percentages in a few weeks, but it was much easier for the manager to enter them 100%, and he had to add them to allow them to report time on the timecard. Therefore we had to require that allocations are locked, and if changes have to be made they have to go through the controller/project. This caused problems when new data entrance screens were made recently because it locked for a lot of other things. People could not report time on projects they were allocated to and so on."

"Allocation of people on a task is causing problems. You cannot enter time on a time card before the notification flow for resource allocation has been completed. (All you want to do is give the department head some turnaround)."

"We (Alfa) are not buying standard items (predefined item numbers) but a unique item. ….. We have defined some high level items that can be used in the workflow authorization, but it is very difficult if the user chooses the wrong item then you have to close the order and start over. …. Because we use unique service items both data and flow in the ERP system is wrong"

"Notification flow required people to be present all the time. The VP trust that we have already made the budget correct thus we could be allowed to accept. Sometimes I call the financial department and make them accept. I understand that an approval is necessary if I change the budget up or down, but moving money from task A to B, not notification flow should be necessary."

"*Allocation of people on a task is causing problems. You cannot enter time on a time card before the notification flow for resource allocation has been completed! -all you want to do is give the department head some turnaround*."

"*We made a process flow (notification flow) where the project manager could do some things and the department manager other things. It is based on roles and there are more roles including managers on different levels. If people don't do their part of the flow, then it all stops. It was well thought trough, but the organization wasn't ready to become so structured (following rules). "*

5.4 Category four: Design issues related to an organizational wide shared taxonomy

This category contains misfits experienced by operational users, controllers and managers relying on data (high data quality and well understood data definitions) to perform their work. Misfits in this category are related to sharing an organizational wide taxonomy. Some of the issues are related to data definitions implemented in the ERP software others to how the data definitions are made operational in the organization.

Some examples given by users:

"*Sometimes I have to enter the same information two or three times. I think ONE is like a picture that is a little unclear, I can see what it is, but it is not clear*."

"*Concepts are not defined clear enough. In the heart of the system it is defined wrong or we have chosen the setup careless. As a result a committed post does not mean the same different places in the system*."

"*The charter of account is too large and complicated, and the customization is causing problems with the original functionality in the system*."

"*We (the IT department) has been forced to make reports where they are able to change the data them self*."

"*First of all finance consider it a financial system and are very offended it is not possible to see all financial data at a snap with the fingers. And the project managers are offended it is not possible to look at the project from the customer perspective. "*

In some cases data definitions and making them operational are interrelated. For example, in one of the interviews an employee in the ERP competence center tells about the difficulties arriving at a design meeting different functional groups needs at the same time, especially a shared definition and interpretation of data. "*….covering three different needs (organizational groups) by the same data input; projects (budget estimate), departments (free resources) and the organization as a whole (management). … Difficult to cover them all at the same time.*" This example relates to the difficulties agreeing on data definitions for resources (employees in projects); data such as skills, different categories of jobs/bookings, the degree to which a booking will be realized etc. As an example the project managers would like to be able to make approximate and uncommitted bookings in order to keep their options open while the department managers want committed and exact bookings to make resource planning more easy and ensure a good utilization of the resources. Another issue related to resources and bookings is the time perspective; project managers and department managers have an operational focus and a short or medium time perspective while (top) management have a strategic (long time) perspective in order to be able to use the booking data to plan the influx and departure of employees with specific skills. When different user groups update and use data according to their specific needs they influence the other groups' possibility of using the data for their different purpose.

Examples of difficulties using data for decision support and financial control:

"*Understanding the report data is difficult - data are not always taken from the place you think.*"

"*And when it comes to data quality then we are not much better off than with EXCEL. You should not have discussions about what costs the system use when making a profit-lost statement, where to get the data in the system. It is not difficult telling what data we want, but it is difficult finding the right place in the system to get them."*

"*Another thing taking very long is to get is a simple contribution margin for the project. This has caused anxiety in the finance department and a very negative attitude. "*

"Cognos are now used for management information. But data quality is still a problem."

"We only use the ERP system to get realized costs. We have almost the same situation as four years ago."

"It is also a problem for project managers on small projects. ONE has a report, but does it show the correct result? There could be an issue about currency, therefore I have to remember to look there ect….."

"Timecards were needed for the financial reports, but other data has been neglected resulting in poor data quality. …..… the timecard (functionality) has been modified to improve the situation"

5.5 Summing up the findings and answering the research question 1.a

The above analysis of the interviews conducted in the user organization and among ERP experts in Alfa's ERP competence center leave an impression of many misfits; situations where the interplay between the factual properties of the IT-artifact and the organizational processes result in:

- Users giving up using the system or introducing serious data quality problems when trying,
- Inappropriate support for work performed by functional groups preventing expedite work processes,
- Ineffective coordination between different user groups, and
- Difficulties using data for financial project control and decision support.

One of the properties of the categories is the different "user groups" or organizational levels related to the misfits. This dimension can be used to investigate who could/should be involved during the implementation process, when to involve them and how. The different misfit categories also focus on different factual properties of the software. All the categories share an element of data modeling but they are distinct in other ways: The first category is focused on the factual properties of the software related to the human computer interface. The second category is focused on the software's built- in perception of the work performed by different functional groups within an organization. The third category focus on coordination between different functional groups, where the coordination mechanisms can be provided/supported by

the software in different ways; simply by sharing data (data model), coded into the original application logic, or developed using work-flow technology on top of the basic application. Finally the factual properties of the software especially important to category four are a shared data model, report generation and "tools" for graphical (ad-hoc) presentation and manipulation of data.

To conclude with an answer to the research question my answer is that the misfits experienced in the user organization after go-live can be understood as belonging to four different categories with different properties. A schematic description of the four categories is given in Table 5.1 below.

Categories' characteristics:	Category 1	Category 2	Category 3	Category 4
Misfit experienced by:	Individual user	Functional groups	Across functional groups	Organization wide
Focus :	Interplay between human and technology	Difficulties performing work as a functional group (professionals)	Coordination mechanisms	Common taxonomy
Factual properties of the software focused at::	Data definitions (data model)			
	Human computer interface – design of user dialogue	Functional application logic: The built-in perception of work performed in functional groups – Tasks to per-form & sequences of tasks	Coordination mechanisms provided by: • simply sharing data • application logic • Work-flow	• Shared data model • Reports / OLAP

Table 5.1: Important properties of the misfit categories.

Chapter 6
How ERP was engaged at Alfa

As the misfits found in chapter 5 illustrate Alfa experienced serious design issues taking their ERP system into use. Many of the issues were not realized until after go-live, and could only be resolved customizing the ERP software. This implies that the design process conducted during phase 2 somehow was defective or that the conditions for design changed over time in the ERP lifecycle. The aim of this chapter is to provide an understanding of how knowledge to decide on the mix of configuration, customizations and organizational change was developed during the implementation process. The analysis of phase 2 is emphasized as this phase is where the major design activity takes place. The other phases are included to provide the context for phase 2, and to illustrate that the conditions for design change over time in the ERP lifecycle.

In this chapter the knowledge integration capabilities of those participating in the design process is analyzed in order to answer the research question:

> How is knowledge to design the system obtained during the ERP experience?

The reminder of this chapter is structured as follows: First a short introduction to the analysis is given in section 6.1. Then a detailed analysis of the different phases in Alfas ERP experience is presented. Alfas ERP experience is divided into four sections: one covering the first two phases (section 6.2), followed by a detailed analysis of the shared insights developed within the functional sub-groups (section 6.3). Then follows a section covering the transition to going live (section 6.4), and finally a section covering the two distinct design phases after going live (section 6.5). In section 6.6 the findings for all five phases are summarized and an answer to the research question is provided.

Readers who aren't interested in all the details in the analysis can skip section 6.2 – 6.5 and go directly to the summarized findings in section 6.6.

6.1 Short introduction to the analysis

The initial interpretation of the empirical material implied that knowledge issues played an important role in the implementation process, especially that the effect of a knowledge breakdown didn't necessary show up immediately. The character of the misfits experienced after go-live imply that many design issues were overlooked or not given sufficient consideration before going live. Thus the analysis conducted in this chapter will provide an understanding of how the design process was engaged in Alfa's ERP lifecycle focusing on how knowledge to decide on the design is developed.

Kensing and Munk-Madsen's (1993) framework (described in more details in chapter 3, e.g. figure 3.2) is used as an analytical tool. However, instead of the two groups, *Users* and *IT professionals* that Kensing and Munk Madsen discuss (1993), I find four different groups of primary actors participating in the design activities throughout the implementation process at

Alfa. User representatives start out on their own during phase one, then in phase two they go together with external consultants, who act as ERP experts until the end of phase two. Thereafter, the external ERP experts leave the scene. During phase 3, some of the user representatives acted as internal ERP experts on behalf of the ERP project team. Their role gets more pronounced, as the external consultants either leave or changed to be employed at Alpha. The further design and implementation now took place between this group of internal ERP experts and the actual users.

To make these changes in actors throughout the ERP lifecycle more visible the analysis is divided into three sections: Section one covering phase 1 and 2, section two covering phase 3 which constituted the transition phase from design to use, and section 3 covering two extensive design phases after go-live, and for each of the three sections the findings is summarized before continuing.

During phase three design is not actually taking place, but the framework is used to illustrate the state of the knowledge base in the organization at the time of go-live.

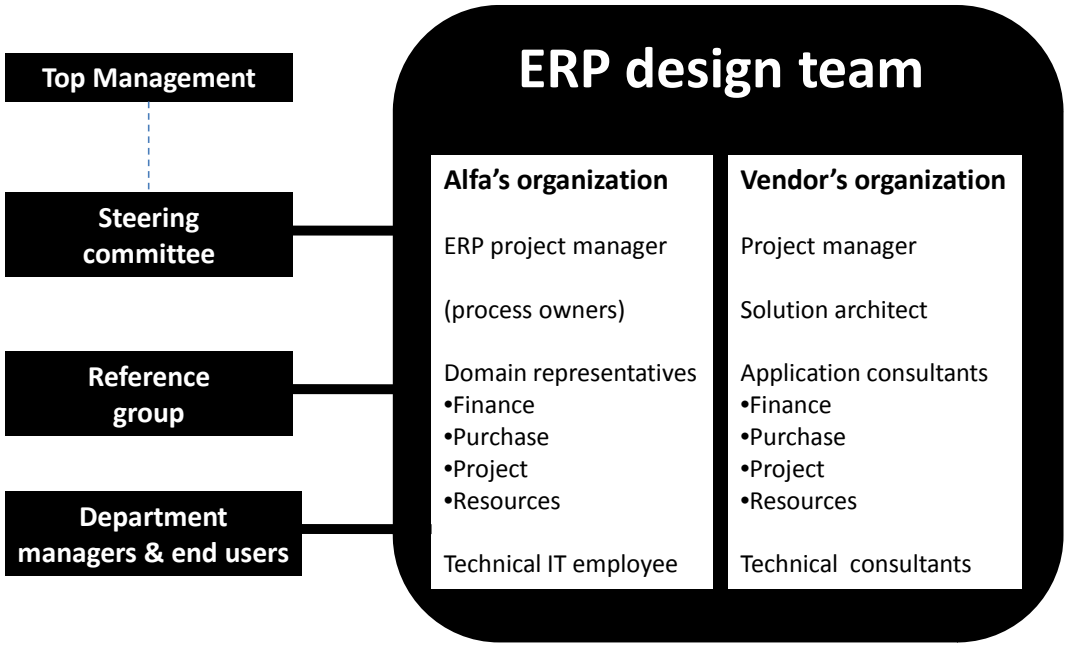


Figure 6.1: Different groups involved in the ERP project

6.2 Section one: Knowledge integration capabilities in phase one and two

As described in chapter 4 Alfa had a project organization for their ERP implementation that had participation from all functional groups within the organization. Some employees were allocated full time to the project and became part of the design team. Some were only allocated part time and were more loosely coupled to the design team.

A group of functional representatives were also appointed to a reference group intended to serve as a backing group, allowing a larger forum for discussing issues of concern coming up in the work performed by the design team.

The design team in Alfa's ERP implementation is a collection of external ERP experts and internal representatives from different functional groups within Alfas organization. Figure 6.1 show in more details the different groups participating in the design team, and the groups formed to support the work of the design team.

6.2.1 User representatives' knowledge integration capability phase one and two

As explained earlier, Alpha provided for a thorough representation of the future users to ensure that the specificities of the different professional practices are supported in the best possible way. People representing all four functional areas within the scope of the ERP implementation were included in the project (the design team) during phase 1 and 2. After the set up of the ERP system some of the user representatives were transferred to a new department; an ERP competence centre.

Phase 1: Requirements specification and systems evaluation

The user representatives met already before the specific ERP vendor was decided. They took part in developing the requirements specification and evaluating the different vendors. Their knowledge integration capabilities during that first phase can be analyzed as follows:

A) Present work abstract-concrete level: The users representatives were employed by Alfa and had extensive practical experience within the functional domain they represented. However, the users' knowledge about cross-functional work processes was limited. As a base for the requirements specification existing work processes were articulated within functional groups using PowerPoint as a tool. The user representatives' ability to integrate knowledge at the abstract-concrete level; relate present practice to the abstract representations, was good.

B) ERP Package abstract-concrete level: The vendor demonstration provided a very shallow abstract understanding of the specific ERP package, and no firsthand practical experience. The user representatives interviewed, indicated, that it was very late that they realized that watching consultants operate the system gave a false impression of the ease of use and the systems' (poor) handling of exceptions and error situations. Furthermore, the users had no or very limited understanding of how different chunks of functionality mutually excluded each other or which derived effect (e.g., across modules) a specific parameter setting would have. Thus knowledge development within the two areas was limited and virtually no bridging of the abstract and concrete level was possible.

C) ERP Package – New system abstract level: Here the written reply to the requirement specification, including the suggested customizations, and the vendor's demo was the only support for knowledge development. Consequently the user representatives' knowledge within the two knowledge areas and their ability to bridge them was very limited

D) Present work – New system: The all-day workshops performed by the vendors gave the users participating an impression of the look and feel of the system as well as the 'chemistry' between the vendor's consultants and Alfa's participants, but very limited knowledge to relate the present work to a possibly new (customized) system. Although a formal evaluation framework was developed, Alfa's participants honestly admitted that the evaluation was based primarily on intuition, as well as the look of the user interface and the interaction with the consultants. Thus during phase 1 the user representatives only developed a very weak idea about the new system and their ability to relate the present practice to a future situation on an operational level was virtually impossible.

E) New system abstract-concrete level: The basis for the new system; the non configured and non customized ERP software was (in theory) available, and so was process diagrams representing the functionality within the ERP package software. Thus you could argue that an abstract representation showing the contour of the new system existed. In practice the complexity of the ERP package made it impossible for user representatives to operate and make sense of the ERP package on their own, and in Alfas case they decided not to make it available to the user representatives in phase 1. The only artifact the user representatives had was the vendors written reply, no organizational specific version of the new system existed. Thus virtually this area was not covered in phase 1.

Phase 2: Configuration and customization of the ERP package

The user representatives participated in the actual scoping, configuration and design of customization in cooperation with the vendor consultants.

A) Present work abstract-concrete level: The requirements specification and some of Alfa's external documents was used in the dialogue with the ERP experts. The process diagrams developed during phase 1 was dismissed by the ERP experts, and no further attempts was made to articulate Alfa's present work in a formalized way was performed, thus bridging the concrete-abstract level of present work were in practice very difficult.

B) ERP Package abstract-concrete level: As a basis for participating in the design activities the user representatives was given some training in the ERP package software. Later design workshops were performed; here the ERP experts demonstrated the capability of the ERP package software and possible Alfa specific solutions was discussed. Pre-defined process diagrams illustrating the work processes implied by the ERP software were supplied with the ERP package software was used during the workshops. These pre-defined process diagrams caused difficulties when used in discussions between user representatives and ERP experts because they were too abstract and open for interpretation, thus they often caused a false perception of common understanding which was only uncovered reviewing the actual instantiation of the processes. Summing up; the user representatives developed some knowledge about the ERP package software and the abstract representations, and as the knowledge increase their ability to bridge the abstract-concrete level also improved although the properties of the notation used cause some difficulties.

C) ERP Package – New system: The process diagrams mentioned above did not provide knowledge about configuration options. Therefore the user representatives had to gain an understanding of the technological options while discussing the options with the ERP experts.

Often the ERP experts would collect information from the user representatives, set up the system, and then come back to show the user representatives the result. Because large parts of the future system could be finalized using configuration the modified (configured) ERP package software served both as an advanced prototype and an emergent finalised design object. Due to the fact that user representatives had gained limited knowledge of the ERP package software, as explained above, they had to rely on the ERP experts to explain the capabilities of the ERP package software and develop design suggestions, their own ability to bridge the capabilities of the software and the design of the new system was limited.

*D) Present work – New system***:** The user representatives very quickly experienced difficulties letting the pre-defined process diagrams guide the mapping of the requirements specification to the new system. As one of the interviewees expressed it:

*"Now everything was **twisted**, we had formulated our requirements based on our business processes but Oracle required us to use their processes configuring the system and designing the new business processes… e.g. in our minds the timing aspects of a project is related to the financial process but in Oracle it is part of the project process".*

Working more or less isolated in functional groups related to a specific module in the ERP package software and using the pre-defined process diagrams made it difficult for user representatives to use their personal experience with present work practice when trying to anticipate how the new processes suggested would work in the organization. Apart from the emergent "prototype" of the ERP software and the requirements specification developed during phase 1 no shared representations of present work or the future system was developed. On an individual basis the user representatives therefore had to find their own way to relate the existing practice to the new system (the design suggestions).

The complexity of the ERP software, especially the cross functional nature (cross module dependencies) made it very difficult for user representatives to bridge the "present work – new system" knowledge areas. Developing test cases and performing the test of the new system provided another opportunity for a reality check. The user representatives however admit that also the test cases to a large extend had a functional (module) focus, and only a limited number of people with specific domain knowledge was involved, and the tests wasn't based on real live data/situations. Furthermore testing customizations did draw much attention, Thus it became more an internal test of the ERP software than an evaluation of the perceived usefulness of the design suggested.

E) New system abstract-concrete level: As described above (B + C) using the pre-defined process diagrams was very difficult, thus they had limited effect mediating the communication during the design process. And Alfa's project manager also acknowledged that they had limited effect as abstract representations of the new processes. The project manager put it this w*ay: "Looking back I can see, that we were wrong assuming that sitting together with the users (user representatives) defining the new processes would make them work in practice …"* . Thus the activities performed and the techniques used during phase 2 constrained the user representatives' ability to bridge the abstract-concrete level.

6.2.2 Consultants' knowledge integration capability phase one and two

In the first two phases, external consultants had the role as the IT professionals in the design, development, and implementation process. Most of the consultants are "*application consultants*" meaning, that they are specialists within a specific module, and know the software's functionality and configuration possibilities seen from the use side rather than the technical side. A senior consultant, *"a solution architect"* with extensive experience implementing the ERP package software and cross-module knowledge was associated with the project and participated on a need basis. Finally technical consultants e.g. database experts or programmers participated in the technical installation of the software and implementing the customizations specified. In this analysis the focus in on the application consultants because they were the once actually participating in the design work perform.

Phase 1: Requirements specification and systems evaluation

Though the consultants were not participating in the preparation of the tender process, they entered the scene when preparing an offer based on the requirements specification.

A) Present work abstract-concrete level: Some of the consultants have practical experience and/or an education related to one of the functional domains covered by the ERP software, but they have no concrete experience from Alfa's organization. Thus during the first phase the requirements specification and a little additional information about Alfa is the only actual knowledge they have about Alfa's work practice. The PowerPoint process diagrams developed as a basis for the requirements specification is not shared with the vendors. Thus the consultants knowledge within the two knowledge areas are very limited, and their ability to bridge between them too.

B) ERP Package abstract-concrete level: The consultants, of course, were familiar with the ERP Package, training material and other internal documentation of the ERP Package, and they had (some) experience implementing the ERP software in other organizations. Thus their ability to bridge between abstract representations of the ERP software and the concrete level was relatively good within the modules where they are experts.

C) ERP Package – New system: Because they knew the ERP package software well and had seen it work in other organizations they had an idea of how it could work. Their perception of Alfa's new system was however only supported by the consultants' interpretation of the requirements specification.

D) Present work – New system: Bridging knowledge from these two areas was only facilitated by the requirements specification during phase 1.

E) New system abstract-concrete level: Due to the limitations in knowledge development explained in C and D the consultants' understanding of the new system was mainly based on their knowledge about the ERP package software. They had not yet developed abstract representations considering Alfa's specific context.

Phase 2: Configuration and customization of the ERP package

After the tender process the actual design and development began. Note that one of the first activities was a re-scoping in order to save part of the costs, which also resulted in a revised requirements specification, fitting with the specific ERP system.

A) Present work abstract-concrete level: The requirements specification and organizational work documents: contracts, invoices, legal information, etc., were the only formalized representations of the present work use during phase 2. In some of the configuration workshops, ad-hoc drawings were constructed around a white board, but seldom preserved. Only one of the four functional groups spent a few hours visiting the related work place, thus, the consultants' firsthand experience with Alfa's work practices was very limited. Consequently, their ability to interpret the (few) abstract representations and communicate with the user representatives about them was to a large extent based on their previous knowledge about similar work processes in other organization.

B) ERP Package abstract-concrete level: See phase 1.

C) ERP Package – New system: Using Oracle's process tool and the pre-defined process diagrams (design proposals) made the consultants relatively comfortable because they could use their existing knowledge about the ERP package software when picturing the new system. Unfortunately, the processes mirrored the module structure of the ERP system and therefore provided limited cross module knowledge development, and at the outset the consultants themselves had limited cross-module knowledge. Thus, in the third iteration (third conference room pilot), a major knowledge breakdown was experienced: Both the finance and the project management consultants expected the other group to provide a solution for a specific requirement, but none of them was able to.

Summing up; within each functional module the application consultants were able to bridge the ERP package – new system knowledge fairly well, but they lacked that capability for cross-module issues and when customizations had to be made.

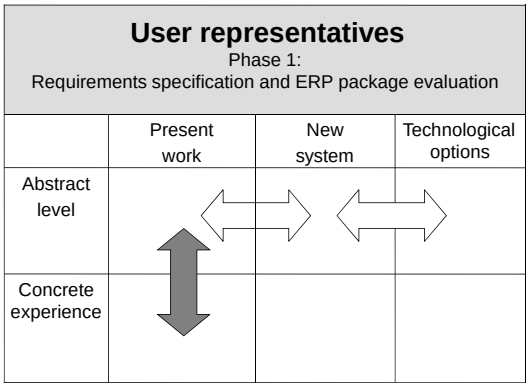
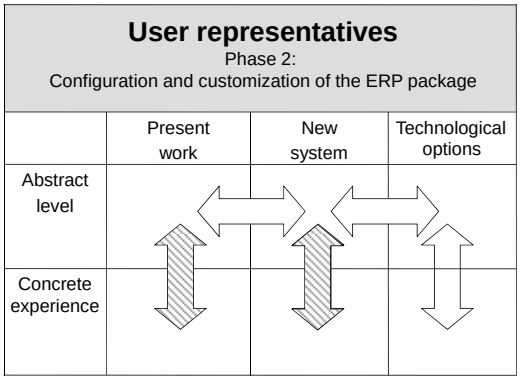
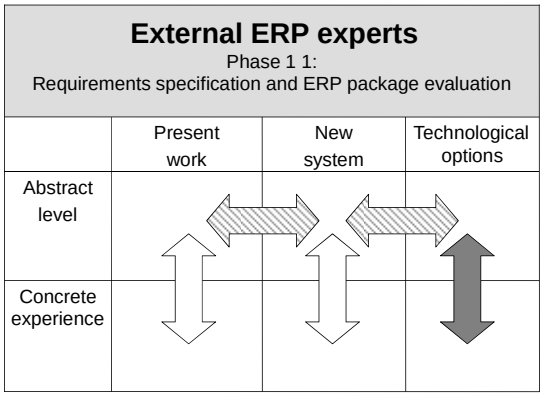
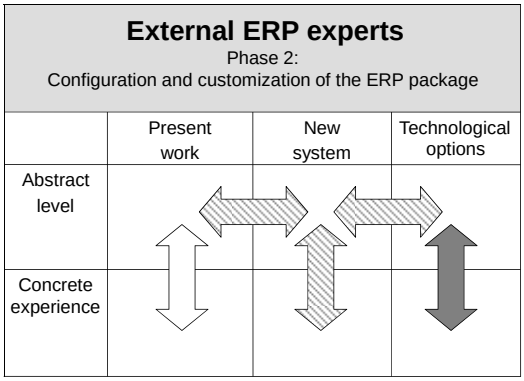
D) Present work – New system: The consultants relied on the users to make the mapping, and as long as the design proposals did not challenge the scope of the project and the users provided the necessary input to configure the ERP system, the consultants were not concerned with the existing or the future work practice.

E) New system abstract-concrete level: The consultants were able to go back and forth between design proposals (processes defined and the configuration documentation) and the configured ERP software. However, they were not able to evaluate the usefulness of the new system in the organization. The requirements specification was still used as a checklist.

6.2.3 Summarizing phase one and two

In table 6.1 a graphical illustration of the knowledge integration capabilities of the actors participating in the design during phase 1 and 2 is provided. Arrows with different filling are used to illustrate the vigor of the integration capability between two knowledge areas: An

empty arrow illustrates a weak ability to bridge the two knowledge areas caused by no or very limited knowledge in one or both knowledge areas; a hatched arrow indicates that some knowledge in both areas was developed, allowing some bridging; and a solid arrow means that extensive knowledge in the two knowledge areas was developed, allowing serviceable bridging.

<table>
<tr><th>Phase 1</th><th>Phase 2</th></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td>Primary design activities:

Specifying processes TO-BE (based on knowledge about AS-IS) in local workshops

Requirements specification is developed within functional groups

Developing a vision for each of the functional groups

Defining project goals related to customers, organizational processes, financial control and employees

Evaluating vendors' demos and written reply to requirements specification</td><td>Primary design activities:

Training the organizational members of the design team in the ERP package software

Workshops in functional groups including external application consultants and organizational members of the design team

Organizational team members discussing with their functional colleges and managers

Information meetings for end users and mangers in Alfa's organization

Two conference room pilots (evaluation of the software design). Only the last one was</td></tr>
</table>

Visiting reference customers The first "Conference room pilot"; 14 days scoping the ERP software to be implemented	cross functional. Developing test material and performing tests Developing training material and performing end user training
Primary design artifacts: PowerPoint slides with the processes for each functional area Requirements specification Vendor's reply to Alfa's requirement specification Evaluation notes for each vendor and final recommendation to steering committee. PowerPoint slides and documents with the vision for each functional area and the goals for the project Scoping document and final contract with the chosen vendor	**Primary design artifacts:** (Oracles implementation template) Process diagrams based on predefined diagrams related to the ERP package software Change requests (customizations or change in scope) Specifications for customizations The ERP software as it moves from being a generic package software towards becoming Alfa 's specific version of the ERP software (configured and customized) Data conversion documents Alfa's "process handbook" Test specifications Training material for end users

Table 6.1: The knowledge integration capabilities of the actors in the design in phase 1 and phase 2, as well as primary design activities and artifacts

6.3 Developing shared insights within functional sub-groups

The process diagrams used were composed by the ERP software vendor and presented as part of the implementation "tool box". The process diagrams are intended to serve as a representation of the business processes (the functionality) implied by the ERP software. The process diagrams are expected to become a shared design artifact mediating both design and construction of the ERP software and the new work processes to be performed in the organization. During scoping the process diagrams are expected to be used to specify the processes to be included in the implementation, during configuration and customization as a

represent of the design object, and during training and use as an abstract representation of both the software and the intended work processes. Thus as an implementation tool the ERP vendor expect the process diagrams to mediate between different phases in the implementation process, between user representatives and ERP experts, and between the design object and the organizational practice.

Looking at the Alfa case and focusing on the *construction dimension* of the process diagrams the level of details provided and the level of formalization became an issue. The process diagrams allow for a high degree of openness and interpretation which caused both user representatives and ERP experts' difficulties:

- Using the diagrams to mediate properties of the ERP software to user representatives seems to induce a false perception of understanding. Thus the people from the use domain only develop a weak understanding of the ERP software's capabilities, and at the same time it made it possible to model something that could not be mediated by the ERP software.
- For the ERP consultants the level of details provided by the process diagrams is also an issue. Thus when used in cooperation with the user representatives the process diagrams do not help facilitate discussing and documenting the necessary details about the user organization in order for the ERP experts to construct the ERP system in a way that reflect the user organizations requirements (needs).

When using the process diagrams as a common representation of properties of the new ERP software it seems too easy for user representatives and ERP experts to arriving at a common (partly false) perception of the "fit" between the organizational practice (the work processes to be) and the properties of the software. The lack of details may be an advantage promoting the ERP software during pre-sale and scoping (phase one) but it seems to be a problem during the configuration (construction) due to the fact that the implementation of the software is very formalized and reflect a specific interpretation of the process diagram.

Another issue related to the construction dimension is that the process diagrams used didn't reflect the configuration possibilities, and as mentioned earlier the construction (configuration) of the ERP software is in part done using configuration parameters. Thus the diagrams did not support design cooperation between different modules/functional groups.

User representatives got frustrated when using Oracles pre-defined process diagrams because they did not reflect their own process descriptions (representations) or their way of working (practice). Thus as a design artifact the process diagrams had limited ability to mediate between practice and design, and between user representatives and designers. Only very late in phase two and during phase three did the user representatives develop sufficient practical experience with the future system allowing them a more life-like interpretation of the process diagrams. The process diagrams however served as an explicit mean of cooperation; they were used as a checklist and tool for reporting progress (degree of completeness).

In Alfa's implementation the (configured) ERP software could be seen as both a design artifact and the design object; it was a continuously moving object of the design activity and at the same time it served as a design artifact mediating the creation of insights and visions into the new information system. As mentioned above a configured version of the ERP

system was used in configuration workshops and that helped the user representatives within a specific functional sub-group get some experience with the future system. Thus the emerging "prototype" of the ERP software had the ability to mediate insight building within the functional sub-groups about the future system although the complexity of the ERP system made it a bit difficult to fully understand the implications of the design.

In the functional sub-groups the participants from both Alfa's organization and the vendors organization in many cases had a common background (education or practical experience) within a specific profession, thus to some extent they could draw on a shared vocabulary and shared theories about the work related to this specific profession.

In the ERP implementation at Alpha, several workshops took place. Especially the conference room pilot and configuration workshops addressed the conception dimension of design work. In Alfa's case however these workshops to some extend had the character of exchanging information. The user representatives were interviewed about the work performed at Alfa, and based on this information the ERP experts would set up the system and show it to the user representatives. This would often be followed by a discussion why the design didn't fit Alfa's way of working, and then the ERP experts would try very hard to explain/convince the user representatives why this way of working would be a useful/possible way for Alfa to work. These workshops provided some mutual understanding, but due to the limited insight into the other part's domain (organization/technology) and the fact that a very complex prototype (that neither of the parties fully understood) was the primary shared representation, only limited mutual empathy were developed. It seems as if it isn't necessary the technique (the workshops around a prototype) in itself that is causing the lack of conception but more likely the spirit in which the technique is used and the interpreted use of the technique that is causing the difficulties.

6.3.1 Develop shared insights across functional sub-groups

Neither the process diagrams nor the prototype (the emerging configuration of the ERP software) were used systematically to mediate insight building between functional sub-groups. Each sub-group focused on their own module/part of processes, and spend almost all their project time working within their own functional area. At the rear occasion they meet to discuss the design, differences in professional backgrounds and lack of knowledge about the other modules in the ERP software constrained the insight building. As the project progressed a few of the user representatives got involved in more sub-groups and then they started to develop cross functional knowledge. This made them able to see relations between functionality (work processes) design in different sub-groups (different software modules).

The misfits found in chapter 5 disclosed that some of the main problems did relate to design of cross-functional, cross-module coordination (work process design). Both, the project ledger and the identity system for procurement items can be seen as coordination mechanisms between the project department, the accountancy, and the procurement department. The configuration workshops mainly focused on business processes within one functional domain neglecting the cross-departmental coordination. Only very late in phase two a work shop with all functional sub-groups was performed.

The vendors architect and the ERP project manager meet now and then to discuss design issues cutting across functional groups; although they tried to inform relevant parties they had no formalized way of sharing their insights with the functional sub-groups.

Writing test cases and performing the test could have been an opportunity to address design issues cutting across functional sub-groups. But again the sub-groups were mainly working individually. The test cases were developed with a functional sub-group perspective and most of the time performed by people related to a specific sub-group. Only very late in the test process did the testers get together in a room to perform specific test sequences that provided insight into how the system performed cross functional coordination.

6.3.2 Developing shared insights for functional sub-group and their peers and managers

In Alfa's ERP implementation insight building is necessary for design (configuration and customization) as well as for assimilation of the system in the organization. As discussed in chapter 5 it wasn't always possible to determine with absolute certainty if a misfit was caused by lack of knowledge about how to operate the ERP software or if Alfa's context left no other possibility than to customize the software. But all the interviewees acknowledged that both issues played an important role. Thus for the functional sub-groups and their peers and managers insight building required two flows of information sharing with opposite directions; insights into the existing practice need to flow from the user organization to the functional sub-group, and insights into the suggested/decided design need to flow from the functional sub-groups to the peers and managers in the organization.

A vision about the new system is developed during phase one of the ERP implementation and involves each of the functional sub-groups. The vision is documented by high level process diagrams, requirements specification, and business goals for the individual functional sub-groups. Thus during phase one three important artifacts are constructed within each of the functional groups facilitating the shared insight building about the current practice and the vision for the future.

The only artifacts carrying over to phase two is the requirements specification and the business goals. Both artifacts are rather abstract and tell very little about the existing practice. The transition to phase two in the ERP implementation mark a change in who is involved in the ERP implementation. The ERP project organization now includes external ERP experts from the chosen vendor (for details about the project organization se section xx above). The requirements specification is kept as an important document, it is the basis for the contract with the vendor (and Alfa's organization) and it is used as a check-list when reporting progress and discussing change requests. In the cooperation between user representatives and ERP experts in the sub-groups much interpretation of the requirements specification is necessary and more detailed has to be added. The requirements specification (as a document) by nature doesn't have the ability to reflect the actual design/document design decisions. Thus neither the requirements specification nor the business goals (documents describing the goals) as such are able to mediate insight building between a functional sub-group and the peers and manager in the organization during the emerging design work in phase two.

The project organization provided some overlap in persons between the functional groups in Alfa's organization and the related sub-groups in the design team. Thus some user representatives worked full time and others part time in the ERP design team. The ERP reference group was not meant to participate in the design work on a daily basis, but to be easily accessible for the design team whenever they needed input/feed-back form the organization. Over time at least some of the user representatives felt discomfort meeting with their peers reporting back from the work in the ERP design team as they felt they had nothing but bad news to their peers.

The process diagrams developed during phase one are discharged and the process diagrams supplied with the ERP software are the basis for developing a process manual for common use in the organization. The diagrams are rather abstract and generic leaving out details necessary to fully understand the implementation of the processes. Thus for people not taking part in the design work in the sub-groups they have very limited ability to inform about the actual implementation unless they are included in discussions/workshops and thereby becoming a boundary object allowing participants form the sub-groups and their peers and managers to develop/share insights. The process diagrams are however never included in discussions/workshops. They are used to make the process owners sign off the implemented design, but since the process owner did not participate in the design work and had no insights into the solution they have to rely on the user representatives in the sub-groups to recommend/back-up signing off the solution. Thus in theory/officially the process owners are backing the solution but in practice they have only a week understanding of the actual implementation.

6.3.3 Developing shared insights at the organizational level

At the beginning of phase one before the actual ERP software was selected management and representatives from all functional areas were involved in articulating the vision for the new system and the derived work processes. Many workshops were performed within functional sub-groups and some cross functional discussions also took place. This was a process taking place at the organizational level in the sense that it involved all functional groups and top management simultaneously. The work was documented by a requirements specification, some high level process flows and business goals for each functional sub-group. The process did however have the weakness that it didn't really address cross functional coordination or taxonomic issues. The visions and the derived requirements developed in each functional sub-group weren't challenged by the other sub-groups; no cross functional perspective was applied.

During phase two no forum was established to develop shared insights at an organizational level.

6.4 Section two: knowledge integration capabilities for phase 3

Phase three constituted the transition from design to use, and the phase where the external consultants leave the organization and the user representatives' role become more pronounced. In this section we meet the user representatives again from the two first phases

again, but in this phase the gradually take over the role of (internal) ERP experts, and some of them developed to become the core of an internal ERP competence centre others return to the user organization as super users.

6.4.1 Internal ERP experts' knowledge integration capability phase three

The transition became visible, when the user representatives took part in introducing the new system and developed training material and documentation for the other users.

A) Present work abstract-concrete level: The abstract representations developed during phase 1 were of no use in this phase, and since no additional representations of the existing work practice had been developed during the design process no shared artifacts articulating the existing practice existed. Thus when communicating with end-users, e.g. explaining the differences between before and now, the internal ERP-experts only had their own experience as employees in Alfa to draw on.

B) ERP Package abstract-concrete level: The internal ERP experts gained more insights into the abstract representations of the ERP package and concrete experience with the ERP package during the development of training material, and when performing end-user training and testing the system. Thus, their ability to bridge the abstract and concrete level increased.

C) ERP Package – New system: The knowledge about the ERP package software and the new system to some extend walked hand in hand, and since the internal ERP experts developed more knowledge within both areas their ability to bridge them also increased.

D) Present work – New system: More knowledge about the factual properties of the new system in addition to spending time with end users; training them, testing functionality and to some extend helping them make sense how to work when using the new system, increased the internal ERP expert's ability to bridge knowledge between present work and the new system. In the communication with end users and when evaluating the design of the new system they could draw on both their personal experience taking part in the development of the new system and their experience working within a department in Alfa.

E) New system abstract-concrete level: As the internal ERP experts participated in developing training material and process documentation, and gaining more hands on experience with the new system their ability to bridge the abstract-concrete level improved.

6.4.2 Users' knowledge integration capability during phase three

As discussed above, the actual users entered the scene again when the user training started and the system was taken into use. Part of the group of user representatives participating in the configuration and customization process moved to an internal ERP competence centre and the rest returned to the user organization. Below the actual end users' knowledge integration capabilities during phase 3 is analyzed.

6.4.3 Phase 3: Training and go-live

During end-user training, it became clear, that some of the defined processes would not work in practice. The result was hostility towards the new system, and some of the planned functionality was taken out just before going live.

A) Present work abstract-concrete level: The implementation approach did not focus on generating shared abstract representation of present work. Further, no cross-functional representations were developed that allowed an integration of knowledge across the organization. Thus virtually this area was not covered.

B) ERP Package abstract-concrete level: As no design was done in this phase, no new abstract knowledge within this area was relevant.

C) ERP Package – New system: See (B).

D) Present work – New system: Given t*he complexity of the new s*ystem, it was difficult for the users to develop an adequate understanding and find ways to perform their work in an effective and efficient way. Additional training, new training material including manual parts of processes, and local documentation of procedures, were developed. But most of the knowledge within this area had to be build trying to use the system. Thus in the beginning of the phase the knowledge integration capability was very limited but it improved somewhat during the phase.

E) New system abstract-concrete level: The new processes documented in Oracle's process tool was suppose to help the end users build knowledge about the functionality in the new ERP system and understand how to perform the new working tasks, they were however never put into use. The end users could not understand them and in many cases the processes depicted in the process diagram could not work in practice. Due to the difficulties using the abstract representations developed the support for understanding the complex ERP system at the concrete level was limited.

6.4.4 Established knowledge base in the organization before go-live

In table 6.2 the knowledge base established in the organization during transition from design to use is graphically illustrated. The external ERP experts are slowly leaving the project and the organizational participants have taken over the role as experts. At the same time the end users have entered the stage. As the graphics illustrate the knowledge base in the organization is during this phase very fragile.

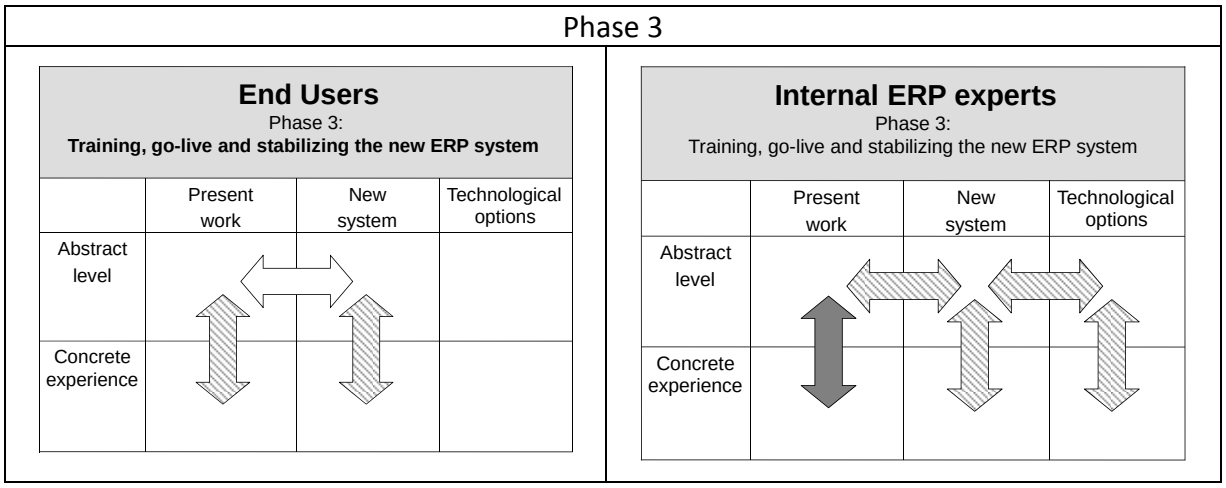


Table 6.2: The knowledge base established in the organization during the transition from design to use

6.5 Section three: Knowledge integration capabilities during phase four and five

Now after going life, the internal ERP competence centre was in charge of supporting the users; collecting feedback about problematic issues and initiating re-design of the configuration and customization of the ERP system. The present work now includes the use of the existing ERP system (developed prior to the actual phase), and when talking about the new system it referrers to a modification of the existing ERP system.

6.5.1 Internal ERP experts knowledge capabilities; phase 4 - the follow-up project

A) Present work abstract-concrete level: Most of the internal ERP experts had worked within functional departments in the organization prior to the ERP implementation, but wasn't involved in the daily work after the introduction of the ERP system. Thus their first hand experience of the present work practice changed and therefore their ability to bridge the abstract-concrete level to some extent decreased. Of course, their prior experience and belonging to a specific functional domain within Alfa was an advantage in relation to communicate with end users.

B) ERP Package abstract-concrete level: By working with Alfa's specific version of the ERP system, the internal ERP-experts also gained insights that could be used to understand the capabilities of the ERP package software. Knowledge that increased their ability to better bridge the abstract-concrete level e.g. cooperating with external ERP experts or interpreting generic system documentation provided by the vendor.

C) ERP Package – New system: Their increased understanding of the ERP package software's capabilities and internal design enable them to develop more qualified design suggestions for new versions of the ERP system.

D) Present work – New system: As already discussed under (A) above the internal ERP experts have the advantage of prior firsthand experience within a functional domain in Alfa's organization thus their ability to communicate with the users and relate to their work context is much better compared to the external consultants.

E) New system abstract-concrete level: The internal ERP experts' ability to bridge the abstract-concrete level improved due to increased experience with the ERP system and the design artifacts used during the development process.

6.5.2 Internal ERP experts' knowledge capabilities; phase 5 - re-design, re-introduction and design of functionality

Another year passed and the re-design process moves into a second iterative circle, now the internal ERP competence centre is well established in their new role.

A) Present work abstract-concrete level: See phase 4

B) ERP Package abstract-concrete level: Over the years, the internal ERP-experts developed a good understanding of the capabilities of the ERP package. Their ability to understand abstract representation like system documentation and communicate with external ERP-experts was well developed.

C) ERP Package – New system: Due to the improved knowledge about the ERP package capabilities, and the internal ERP experts experience with ERP design work their ability to develop and evaluate design suggestions improved dramatically over the years.

D) Present work – New system: Informal user networks in the organization were formed and some of the internal ERP experts were included. Within these networks knowledge about the ERP system was shared and better solutions to actual difficulties using the ERP system was discussed. As a result important cross-functional process knowledge was developed. Thus engaging so close with the users and at the same expanding their knowledge about the ERP system's capabilities resulted in better-informed design suggestions; design suggestions that better meet the needs of the organization.

E) New system abstract-concrete level: The internal ERP experts sometimes experienced difficulties "predicting" how the new system would work (be received) in the organization. Thus, some functionality was re-designed several times without achieving real success, but in general new or re-designed functionality was adopted more easily in the organization.

6.5.3 The user's (end user representatives) knowledge integration capabilities; phase 4 - the follow-up project

The users start in cooperation with the internal ERP experts to make things work. Now the practical experience with the ERP system facilitates the learning process.

A) Present work abstract-concrete level: Ad-hoc material is developed helping to facilitate communication within and across functional departments.

B) ERP Package abstract-concrete level: Practical experience with the ERP system improved the user's ability to understand abstract representations, but due to the complexity of the system, users experienced difficulties understanding how local changes would affect other parts of the system. Thus their understanding of "technological options" and derived consequences when modifying something only improved a little.

C) ERP Package – New system: Practical experience with the ERP system helped users explore the ERP package functionality on their own. Their ability to understand and contribute with design suggestions improved accordingly.

D) Present work – New system: Experience with the ERP system improved the ability to envision how re-designed or new functionality would work. Nonetheless, to fully understand the consequences of a design proposal before going-live remained a challenge.

E) New system abstract-concrete level: Additional training in the use of the new system was given. Users helped each other, that is, they developed and shared small representations of important functionality. Some of the users who had participated in the configuration and customization contributed official 'quick guides' explaining how to use the new software step by step. However, the complexity of the new system caused difficulties for the users to generate sufficient abstract knowledge regarding dependencies between different parts of the system.

6.5.4 The user's (end user representatives) knowledge integration capabilities; phase 5 - re-design, re-introduction and design of functionality

A) Present work abstract-concrete level: Over time very detailed user manuals were developed. Thus, in familiar areas the users had good possibilities of bridging the concrete and abstract levels related to the use of the ERP system. In general, the users did not engage in developing or interpreting codified abstract representations of the post implementation work processes. No common tool or technique was used to support the general development of codified abstract representations. Some ad-hoc groups worked at improving the use of the system and/or re-design of functionality. In case re-design of ERP functionality was needed, the users were requested to use a specific template when specifying requirements. In the controlling department, users worked systematically to develop a controlling manual (process documentation) in order to provide a common understanding among controllers in Alfa about what to do and how to use the ERP system and other tools. This documentation exposed flaws in the design of the ERP system which resulted in poor data quality. Unresolved taxonomic issues related to provide comparability, e.g., across project and over time, were discovered.

B) ERP Package abstract-concrete level: In general, the users did not explore the standard version of the ERP software but focused on the configured and customized software. But users engaging in re-design of the ERP software or trying to understand the implications of changing data definitions or data semantics expressed great difficulties in understanding the complexity of the software.

C) ERP Package – New system: See B.

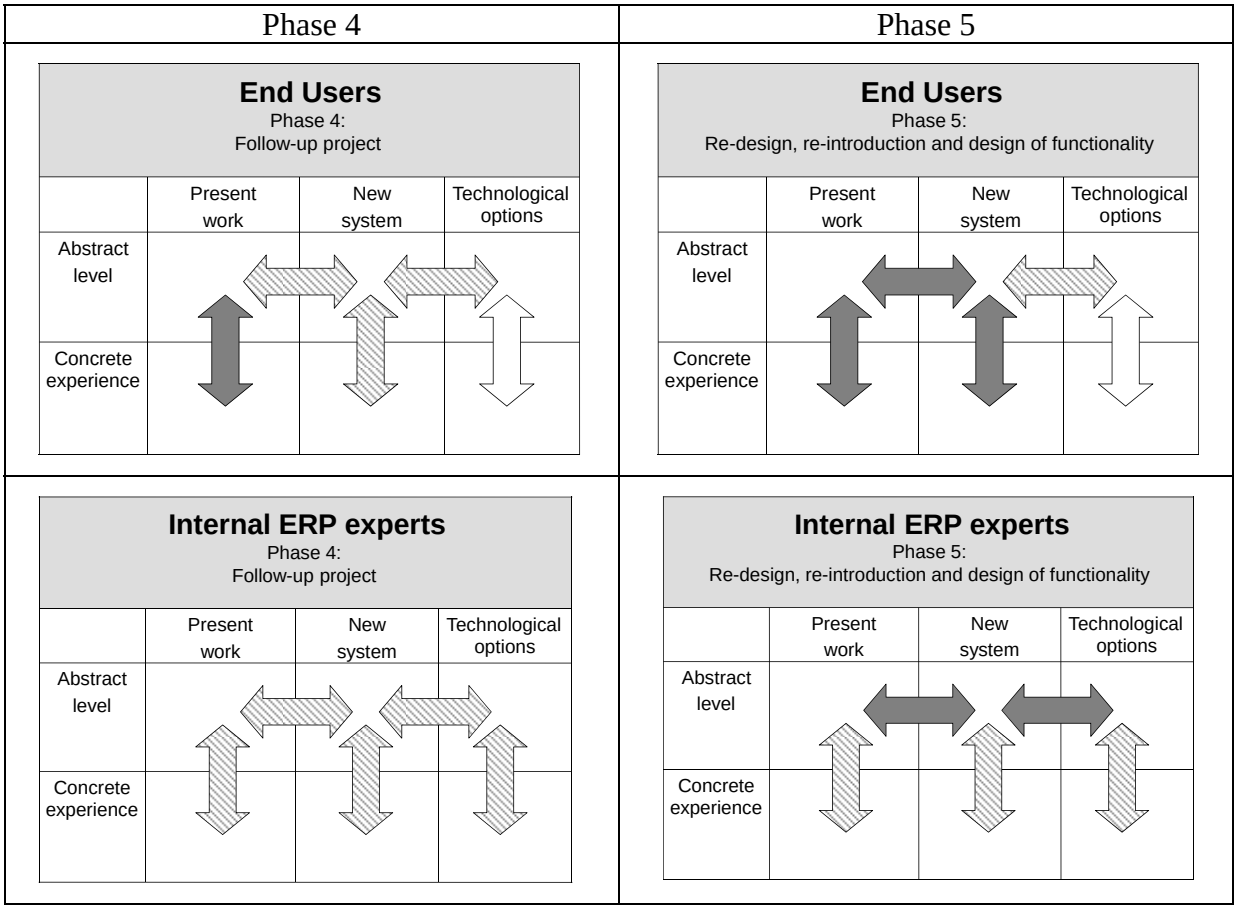
D) Present work – New system: Some users only developed a fragile understanding of how to use the ERP system for their present work. But informal networks of employees were formed, providing a well developed cross-functional understanding of how to use the system. Users engaged in re-design of the software were typically part of such an informal network and had a good understanding of the software and cross-functional work practice.

E) New system abstract-concrete level: Alfa's users still experienced some difficulties evaluating design suggestions. Especially when they involved customizations to be programmed externally.

6.5.5 Summarizing phase 4 and 5 obtaining knowledge for design

After go-live both end-users and internal ERP experts get practical experience with the ERP software and using it in the organization, thus the knowledge base changing dramatically. They have shared experiences and to some extend a shared vocabulary.

In Table 6.3 is given an overview of the knowledge integration capabilities of the actors participating during phase 4 and 5.



Activities: When using the system individuals and functional groups find ways to change the socio-technical design – design in use	**Activities:** Projects are initiated considering how to improve the use of the system and especially developing a shared taxonomic e.g. project establishing a project controller manual.
Artifacts: Additional training material A list of more than 500 issues are developed Change requests & specifications of customization Ad-hoc material are developed in the user organization documenting how to work (use the system)	**Artifacts:** Controller manual

Table 6.3: The knowledge base established in the organization during the transition from design to use

6.6 Conclusion: answering the research question 1.b

The aim of this chapter is to provide an understanding how knowledge to decide on the design of the new ERP system is obtained. Thus it provides an answer to the research question:

How is knowledge to design the system obtained during the ERP experience?

The analysis shows that knowledge is obtained in very different ways at different phases in the ERP lifecycle.

In the first phase the organization go through a process of investigating existing work processes in order to obtain an understanding of requirements they need future ERP package software to accommodate, and to develop a vision of future (desirable) work processes. A number of business cases are formulated to support the vision. Most of the work is performed in functional groups, and as a result the requirements, the vision for future processes and the business cases are related to the functional groups. At the end of phase one cursory knowledge about candidate ERP package software (the technological options) is obtained through 2 days workshops performed by the vendor based on material (requirement specification) provided by Alfa. During phase one the existing practice is in focus and the organization is in charge of the process including all organizational levels in the process.

In phase two the design work is performed within a design team organized in four functional sub-groups. Only very limited work is performed across the functional groups and only very

late in phase two. The process diagrams developed during the first phase is dismissed and no other abstractions of the present work processes are developed. The primary design artifacts used are the requirement specification, abstract process diagrams provided by the vendor (somewhat reflecting work processes provided by the ERP package software), and the ERP package software used as a prototype. The way the design work is performed it has a didactic nature; the organizational representatives are being taught by the ERP experts.

Within the functional groups the process doesn't provide much support for achieving new understanding of the existing organizational situation, or conceiving ideas about the future (how and why to work differently/change). Such knowledge has to be developed as a side effect of discussing with the external ERP expert if work processes demonstrated (using the ERP software as a prototype) can be used in the organization. The focus is on the ERP package software, and the design process is dominated by the external ERP experts. During phase two neither the ERP experts nor the user representatives develop knowledge within all six knowledge areas (Kensing and Munk-Madsen 1993). In that regard phase two has two important weaknesses: First, the external IT experts used a design strategy that involved user representatives as informants only. As the field material indicate, it was not possible for the user representatives to relate their knowledge well enough to the design proposals. This weakness resulted in several serious design issues.

Second, I found a lack of cross-functional cross-module knowledge integration, both among user representatives and external ERP experts. A configuration workshop including all functional sub-groups was performed shortly before go-live. The workshop brought up several cross-functional design issues at this very late stage. One of the issues related to joined work between accountancy and project management around project budgets was so serious that it almost closed down the project - a major customization was necessary. Prior to the workshop, the external ERP-consultants in both the finance group and the project management group had assured the user representatives that the functionality was provided in the other module.

Phase 3 constituted the transition from design to use. In Alfa's case end-user training and testing allowed many end-users in the organization to be acquainted with the ERP software for the first time, and the reaction wasn't positive. Especially the project managers (responsible for construction projects some amounting up to a 100s of millions of Danish kroner) are not satisfied with the functionality provided by the system. They managed to get large chunks of functionality dismissed before go-live, and after go-live they are able to convince management that serious customizations is necessary in order to make simple functionality use full. Going live the end-user organization lack knowledge how to operate the system, they either give up completely or introduced data quality problems that made it difficult to make even simple functionality work for other user groups that depended on these data to be correct.

However in phase 4 and 5, the knowledge integration capability has changed dramatically. Thus when re-design is requested (re-configuration, re-definition of data or customizations) each participant covers more knowledge areas, and the communication between users and ERP experts is supported by shared experiences and a shared vocabulary. However the internal ERP Competence Centre still required change requests to be initiated by a requirements specification and a business case. Practical experience with the system and

informal cross-functional networks in the organization, however, provided insights that allowed a cooperative design approach. It is also obvious that the rhetoric used during phase two: “you need to adapt the work processes to the systems way of working” has changed, instead serious considerations how to make the socio-technical design support the business goals takes place. E.g. it is important to get hours spend on projects reported timely and correct, thus if the human computer interface for *time sheets* (reporting hours spend on project tasks) is the reason for the difficulties then it needs to be changed.

Chapter 7
Concluding on the first learning cycle

In this chapter I will summarize the result of the first learning cycle and provide an understanding for the work performed in the next learning cycle.

7.1 What did the first learning cycle contribute (what do we know now – where 'we' is me and the reader)?

- We know more about what constitute misfits seen form the user organizations perspective (unresolved design issues carrying over from the initial design phase to use). A framework with four misfit categories is developed, the framework show that misfits are found on many different levels in the organization. Existing literature seems to recognize especially the process view, but does not necessarily perceive it as a problem of design.
- We know that the design process as it was conducted during phase two resulted in these misfits, and that the design process and the derived design were meet with rejection and resistance in the user organization
- We also know that a considerable amount of misfits were caused by the factual properties of the ERP software e.g. Alfa using service items with a very special character related to the construction industry, another example is the need for a project hierarchy instead of a managerial hierarchy in the resource allocation workflow. Thus Alfas initial presumption that some customizations would be necessary due to their specific organizational context seems to be confirmed.
- We know that in practice Alfa's ERP implementation has a duality in the design object, and that it seems reasonable to perceive the implementation process (deciding on the mix of configuration, customization and organizational change) as a design process instead of simply as a process of populating package software with organizational specific data.
- We know that phase two did not provide sufficient knowledge integration within the design team for design, and that the design artifacts used focused on explicit means for cooperation between the vendor and the client organization, lacking the ability to provide sufficient shared experiences, insights and vision about the socio-technical design. Especially the user representatives from purchasing intuitively felt that their
- peers would be very disappointed (the same seems to be true for the project managers), thus he felt real discomfort informing his peers about the new system during phase two.
- We know that Alfa accepted to follow the implementation method suggested (recommended) by the vendor, and that the user organization consciously or unconsciously allowed the consultants to dominate the design process and the ERP package software to constrain the design space. The way the process were conducted during phase two it did not allow engaging the organizational members outside the design team, thus no wider commitment to or accept of the realized design was established, and limited knowledge in the user organization about what to come.
- We know that the condition for design changed during phase 4 and 5, and that many of the design issues were resolved – especially many of those related to the human computer interface (customizations were made). And taxonomic issues had gained

organizational attention especially driven by a stakeholder that could mediate between project managers, internal ERP experts and management namely the project controllers.
- We also know that many of the customizations around project functionality that were specified (and paid for) by Alfa also were requested by other project oriented organizations, and at that the vendor later decided to include these customizations in new versions of the package software.

7.2 A preliminary answer to the thesis overall research question

The overall research question for this thesis is: How do organizations engage ERP implementations and why does it often result in misfits? Based on the findings provided by the first learning cycle a preliminary answer to the overall research question would be:

The ERP implementation is engaged as a series of design phases (in Alfas case e.g. phase 2, 4, 5, and more to come). One wide ranging design episode is conducted before go-live and a continuous series of smaller design episodes take place after go-live. The conditions for the design work seem to differ considerably before and after go-live.

Before go-live the implementing organization depend on external experts to provide knowledge about the technological options and the skills to configure the software. The design process is constrained by time and budget issues, the actual design work are dominated by the consultants, and the design space is constrained by the capabilities of the ERP package software. The primary design artifacts used to mediate the design work is process diagrams and the ERP software used as an emerging prototype.

The misfit categories serve to exemplify the design issues and the knowledge breakdowns in the implementation process as it is approached in Alfas case. Thus the design artifacts used in the implementation process lacked the ability to form a base where user representatives and external consultants could discuss the design and the changes implied by the suggested design of the ERP system. This is especially critical as it means that the design and the derived changes haven't been negotiated (committed) in the organization prior to go-live.

After go-live all the participants in the design work is internal to the implementing organization. Practical experience with the ERP system changes the knowledge base for design activities, thus both internal ERP experts and user representatives cover more knowledge areas (concrete knowledge with the present work and concrete experience with the technology), and the communication between internal ERP experts and users is supported by shared experiences and a shared vocabulary. The conditions for design work (re-design) has changed dramatically although the ERP competence center still take on the responsibility ensuring that as few customizations as possible are made to the ERP software. Now the focus is on making the organization use the system, and using it in a way that provide a data quality that allow important (sometimes even automated) decisions to be based on the data. Thus organizational issues, use issues (the organizational practice) are now at the center of design discussions. Time and budget constraints can no longer be used as an excuse to dismiss serious concerns form the user organization. The conditions and the knowledge base for design related to cross functional coordination and taxonomic issues have also changed dramatically, thus informal cross-functional networks in the organization provide an

unofficial forum for exploring and negotiating different stakeholders' needs and point of views.

7.3 How does the misfits relate to the implementation approach?

The amount of design issues Alfa experienced indicate that the introduction of the new ERP system in the organization isn't simply a matter of populating the ERP software with data and putting it into effect. The serious issues regarding the factual properties of the ERP software also indicate that the ERP software has to be perceived as a semi-finished product that needs to undergo a socio-technical design process before it will become useful in the organization. Looking at Alphas ERP implementation cycle socio-technical design issues seems to have received very little attention during phase two. This seems to instigate the user organization to revolt against the ERP system during phase three and four.

The misfit categories found indicate that the socio-technical design process should involve many heterogeneous groups of actors and requires simultaneous socio-technical design on different organizational levels including developing a shared taxonomy; shared understanding of concepts (data) needed for operational purposes as well as decision support. Reflecting on how the socio-technical issues experienced relate to the approach used in Alfas ERP implementation phase two it seems to be problematic.

The importance of the human computer interaction is not realized during the first two phases in the ERP implementation. Why not? The vast amount of customizations performed after going live support the importance of addressing this issue. The difficulties experienced in Alfas organization resulted in data quality problems that it took very long time to recover from.

Most of the design work performed during phase 2 had a focus on a specific functional group/professional group. The modular construction of the ERP software supported this focus. And the fact that the external ERP expert's skills are built around modules intensifies it. The process diagrams were used primary within functional groups, and test material and training material also focused on a specific module. Thus design work addressing misfit category three and four (cross functional coordination and developing a shared taxonomic) received very little attention. Why? As Alfa as an organization had very limited experience with integrated IT systems they might not have been aware of the importance of cross functional design activities, and the approach used by the external ERP experts did not support it.

However, although the approach used had a focus on functional groups the misfit analysis also uncover several design issues related to work within functional groups. Why were they not discovered/resolved during the initial design in phase two? To be fair some of them actually was discovered but could not be resolved in a satisfying way e.g. the ERP package software did not allow two different principles for acknowledging the receipt of goods and services at the same time. But a serious number of issues were not discovered until going live!

The analysis of the implementation approach used in Alfa's ERP implementation shows that, intentionally or not, the design team seems to be isolated during phase two; focusing on

insight building within the design team. Furthermore it seems to privilege a focus on the construction (configuration and customization) of the ERP software, especially the external ERP experts' work/responsibilities in the construction process. Thus the design artifacts used focus on explicit means for cooperation between user representatives (the implementing organization) and the external ERP experts (the ERP vendor). Very limited attention is given to sharing experiences, insight and visions about the socio-technical design. The way the process was performed it had a more didactical nature; the user representatives being taught by the ERP experts. It did not provide much support for achieving new understanding of the existing (organizational situation) or conceiving ideas about the future (how and why to work differently/change work processes).

7.4 How does it relate to what is already known by the literature?

At the surface Alfa followed an approach similar to a more widely expected ERP lifecycle (Bancroft, Seip et al. 1998; Markus and Tanis 2000); Thorough evaluation of potential candidate ERP package software, organizational consensus about the ERP package software chosen, the implementation approach and the plans for user involvement is widely accepted in the organization, applying the vendors' method, using external consultants etc. Alfa was very conscious about having users participate, the design team was balanced and cross-functional, and both business knowledge and technical knowledge (ERP expertise) was present in the design team. However, Alfa did not manage to achieve the necessary knowledge integration; neither for design nor for assimilation of the new system in the organization.

Thus the case study confirms Robey et al's (2002) findings that knowledge integration within the design team is important. However the approach used in Alfas case doesn't result in successful communication between the ERP experts and the organizational representatives. Social bonding isn't achieved, at least not in the sense that the external ERP consultants care to understand the existing organizational practice at a level where they are able to understand organizational concerns and help the organization evaluate design suggestions.

In the PD tradition, representation isn't discussed as a way to capture knowledge, but as a base for communication around design. Representations are understood as design artifacts that mediate communication between different professional groups (Kensing and Blomberg 1998; Bertelsen 2001). The late discovery of mismatches between the planned implementation and the Alfa's specific requirements indicate that knowledge about the work practices, the technical possibilities and the specific implementation were not integrated well enough.

Huang and Newell (2003) found that it is necessary to engage organizational members through the promotion of project benefits, and manage social networks in order to make the ERP system successful (accepted) in the organization. Alfas approach during phase two however made it problematic to achieve this kind of knowledge integration. Thus some of the user representatives participating in the design work admitt that they were reluctant to share news about the ERP software with their peers during phase two because they felt that they had nothing but "bad news". The user organizations success having functionality dismissed and being allowed to resist using the system indicate that the new system didn't have much support within the organization. After go-live (years after) different groups in the user organization started to realize (on their own) how to benefit from using (and maybe changing)

the ERP software, and they starting promoting the ERP system as a mean to achieve the benefits. As more people realized how to benefit from having the system used throughout the organization, and many design issues were resolved, then the general attitude toward the system changed somewhat and the resistance slowly changed toward a more positive attitude. Thus the case seems to confirm that realizing how to benefit from the ERP system and promoting the benefits affect the user organizations attitude toward the ERP system and their personal involvement in the design work. In Alfa's case this situation is however not achieved until years after go-live.

Although Alfa started out developing business cases, a vision for the new system, and high level process diagrams, it turned out that these artifacts became impossible to use actively during phase two. Alfa also appointed process owners (managers in the organization), used the vendor's process tool and developed a process handbook for all processes; anyway they ended up with an approach lacking a cross organizational process perspective, and they were unable to promote benefits. How can this be explained? First of all the primary artifact regulating the scope and cooperation was the requirements specification not business cases or the vision. The consultants were dominating the design and implementing process before go live, and they had an incentive to fulfill the requirements specification, but not ensure that the solution would "fit" the organization, or ensure that the design was negotiated and accepted in the organization. Secondary the ERP experts' skills and knowledge was related to modules in the ERP software, and the design work wasn't organized in a way that allowed the functional groups to develop shared insights. Third the approach used made it difficult for the organizational representatives to engage the reminder of the organization. The approach had a focus on the functionality provided by the ERP package software.

Hence although Alfa (maybe a bit naïve) assumed they had a project setup that followed "best practice" ERP package software implementation they in several aspects ended up with an approach neglecting some of the known risk factors. The project was successful in the sense that it was on time and budget, but they were unable to foresee where organizational changes were necessary to make the design work and they design of the new system lacked support (commitment) at all levels in the organization.

7.5 Where is the research going?

From here the research could have gone in many different directions. I could have included management; I could have conducted more interviews trying to investigate in more details why a cross-functional process perspective was lacking; I could also have focused on how the organization continued to re-design processes and the ERP software over the years. However, I decided to stick to the focus on the people actually engaged in the practical design work, I was interested in understanding in more details how the consultants and the user organization actually develop shared knowledge in order to decide on the mix of configuration, customization and organizational change in the design phase before go-live.

I was wondering why knowledge about the existing organizational practice had very little priority for the consultants, but at the same time user participation is perceived to be a critical success factor in ERP implementations. The consultants participating in Alfas implementation

seemed to use an approach that is widely agreed on by ERP vendors and ERP consultancy companies, why do they consider it an appropriate approach?

I was also wondering why the consultants chose to use the design artifacts they did, e.g. no abstract formal representations, but instead used the ERP software as the primary artifact facilitating discussions with user representatives? Thus I was interested in understanding if the approach used in the case organization was a more general approach, and if so how consultants make sense of using this approach.

The case study also made it very clear that ERP implementations are not just about populating a standard system with organizational data. It seems to be a design process with dual design objects (the IT artifact and the organizational processes). Thus I was also wondering how the duality in the design object (simultaneous and intertwined design of the two design objects) could be understood at a theoretical level?

Thus the findings from the first learning cycle pushed me in two different directions. First of all to investigating the rationale behind the approach used seen from the ERP experts' perspective (addressed in part III), and second to reflect on how to understand (at a theoretical level) the implementation process and the design issues if considering the duality in design object (this is addressed in part IV).

.

In the method section I promised to let the reader in on my personal bias and therefore it is fair to admit, that deciding on where to go after the first learning cycle was based on my personal interest, not any kind of objective evaluation of what had more relevance. The background for this is that I am brought up in a socio-technical tradition (Avison and Wood-Harper 1990; Mumford 2003) at the Copenhagen Business School in the late 1980s. This background had made me skeptical toward the approach used. I believe all stakeholders should be considered and included when developing and implementing Information Systems in organizations.

Part III - The second learning cycle

Part III - *The second learning cycle* contains three chapters. In Chapter 8 I discuss and present the research method for the second learning cycle. Then Chapter 9 provides the results from the focus group study; the answers to the three detailed research questions (2a, 2b and 2c). Finally, in chapter 10 I conclude on the second learning cycle.

In this third part of the thesis I focus on the ERP professionals' (external and internal ERP experts) and their perception of ERP implementations, and strive to understand how this perception influences their relation to the user organization and their preferences in tools and techniques used during the cooperative design process.

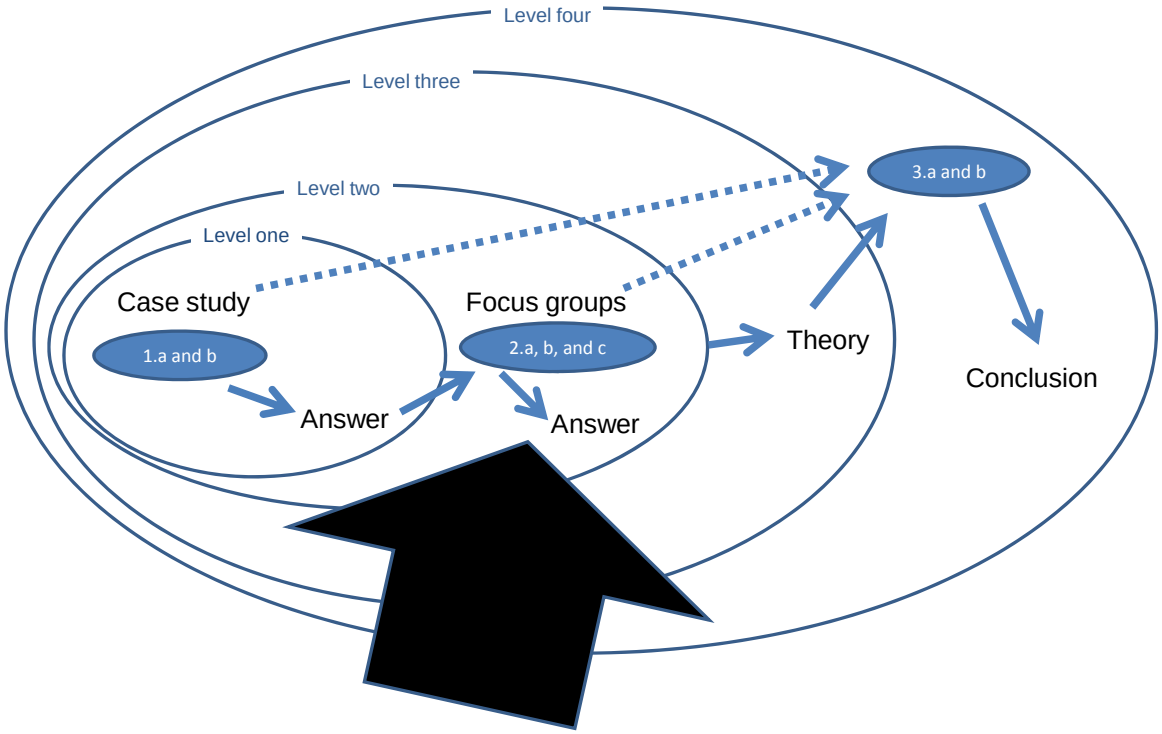


Figure III.1: The overall structure of the Ph.D. thesis with a thick arrow indicating where part III belongs

Chapter 8
Research Method for the 2nd learning cycle

In this chapter I discuss and present the research method for the second learning cycle

My next level of realization is related to a focus group study. The insights developed during the first learning cycle gave rise to new and more detailed questions. Why were the ERP experts dominating the design process? Why were traditional design artifacts providing insights about the use domain not used? Why was knowledge about the use context almost absent during the initial design process? Why were the design of cross functional coordination not a natural part of the external ERP experts methodology?

As explained in the previous chapter I decided to investigate how the larger context of ERP implementations might have influenced how the ERP implementation was approached in the case organization, and I decided to focus on the ERP professionals helping organizations implement the package software.

Thus the second learning cycle is intended to provide an understanding of how ERP professionals perceive ERP implementations, and establish an understanding of the relation between their understanding of ERP implementations and the way they approach the cooperation with the user organization; both their attitude toward the user organization and the techniques and tools used to mediate the cooperation.

Three new research questions for the focus group study were formulated:

2a) How do ERP professionals perceive ERP implementations; what are they about?
2b) How do ERP professionals perceive the need for knowledge integration in order to design the ERP system?
2c) Which design artifacts do ERP professionals use and why?

The reminder of this chapter is organized as follows (I need more sections)

8.1 Research design, 2nd learning cycle

The use of focus groups within marketing and social science can be traced back to the 1930s but has really taken off during the 1980s and 1990s as a way of collecting qualitative data (Morgan 1997). I have not met much research on Information Systems using focus groups, but never the less I have chosen to use the technique because it is able to contribute something unique to my research project and my understanding of ERP implementations. Using focus groups as a research technique allow me to collect data in a relatively time efficient way through group interaction on a topic determined by me. A focus group cannot substitute either individual interviews or participant observations, but it can provide access to a form of data that are not easily obtained by either approach(Morgan 1997). My main arguments for using focus groups are:

- The ability to bring consultants and uses together outside a specific project discussing and sharing insights about ERP implementations. Insights and issues that a customer-client relationship in a specific project would make very difficult to talk open about.
- The opportunity to observe a large amount of interaction on a topic chosen by me in a limited period of time (given my ability to assemble and direct the focus group sessions)
- To provide data about attitudes and decision making related to the implementation process and the techniques and tools used (developed while the participants argue their point in a group discussion); learning about the participants experiences and perspectives.
- Discuss agreements and disagreements among peers provide direct access to similarities and differences in the participants' opinions and experiences, and may also trick tacit knowledge
- Provide insights into the common (socially constructed) understanding of ERP implementations as a practice
- Finally it provides an opportunity to investigate if my interpretations of what is going on in the in-depth case study can be supported.

The themes the focus groups are expected to provide data about are:

1. The anticipated usefulness of different techniques and tools to facilitate knowledge integration during the design process performed before go-live
2. Attitudes and approaches to involve the users (user representatives as well as end users) in the design process
3. If the ERP-professionals' understanding of the nature of an ERP implementation and their understanding of their own role in an ERP implementation can explain the implementation approach used in the in-depth case study

Assembling the focus groups I had to balance many different requirements and practical obstacles. The participants should have solid practical experience implementing ERP package software and be willing to spend 4 hours at the university. Preferable they should come from different organizations and together cover at least a small number of different but well known ERP packages. My aim was to have professional ERP implementers (from vendors or consultant companies) as well as organizational representatives who had participated in an ERP implementation in their organization. The number of focus groups and the number of participants in each group also had to be decided on. Doing qualitative interpretive research one group may be sufficient. For example Wagner (2002) says (p. 255):

"The rich descriptions obtained from process data are useful in themselves, but we argued in this dissertation that the scale is not an issue when trying to gain theoretical insight. From a Latourian perspective the grand narrative and the individual stories illuminate one another."

But since I had limited experience moderating focus groups and I could not be sure it would work out the first time I decided to aim for four groups with 5-6 participants. If I could get enough volunteers I might have the opportunity to form both homogeneous and heterogeneous groups (only consultants group, only organizational representatives group and mixed groups).

Finding the participants I decided to use my network's network (as recommended by Bente Halkier (2002)). I graduated from Copenhagen Business School in 1991 and knew that many of the "class mates" worked as ERP consultants or knew someone who did. I therefore wrote them and asked if they would pass on an invitation in their network. I got 15 positive responses and decided to aim for 3 groups with 5 participants each. Finding a meeting time for the focus groups that suited the volunteers was not easy, thus the groups were more or less put together according to available time slots in their calendar. Therefore all groups ended up being mixed groups. Due to influenza epidemic at the time for conducting the focus groups one of the groups had to be canceled and it turned out to be impossible to reschedule it. Thus I ended up with two successful focus groups.

Preparing the focus groups my own role as a moderator had to be considered carefully. In focus group research the researcher (moderator) is using him/her self as a methodological tool (Morgan 1997). The researcher decides who is participating and what to be discussed, and afterwards the interaction in the group is interpreted by the researcher. Thus the researcher creates the social space where the participants experiences and interpretations can be expressed in a dialog with the moderator; a co-production of research material.

During the focus group session the moderator need to focus on facilitating the discussion and handle the social dynamic in the group making sure everybody is heard. Listening is important, listening and asking question, balancing the understanding and the distance letting the participants talk (Morgan 1997). Being ignorant when asking clarifying questions can be very difficult for someone familiar with the subject. And maybe even more difficult if the participants expect you to know a lot on the subject on beforehand. But as I was very open with the focus groups about my history as a professional ERP consultant (some of them knew anyway so there was no point in trying to keep it a secret) I tried to make it clear, that I intended to stay out of the discussion and maybe even ask "stupid" questions if needed to clarify something although they might expect me to be able to answer the question myself. In the introduction to the focus groups I also made it clear that it was their discussion that was valuable to me, not that I immediately understood or agreed with what they were saying; I explained that I was there to learn from them and what was important was their experience "documented" through the discussions.

Another consideration planning the focus groups had to do with the amount of structure and moderation used in the focus group. The literature suggests three different models (Morgan 1997):

1. A loose structured model with very broad and few questions starting with some open questions. Used for explorative data production and pre-analysis for survey etc. Or if you are more interested in listening to the participants discuss with each others.
2. A very structured model with more and more specific questions and maybe and some exercises. If focus is more on the matter of the discussion then the interaction between the participants. Or explorative if you want many different questions investigated. Best if the moderator/researcher has extensive knowledge about the subject, and therefore wants to have very specific questions answered. However it means that it can be difficult to get new knowledge.
3. The combined funnel model where you start out with open questions and end more structure. This will allow space for an open discussion and at the same time ensure that

specific issues are discussed. Decisions about contend should be made according to the desired/planed knowledge generation. The combined funnel model suggests open questions to start with inspired by e.g. ethnographical findings. Later more specific questions and follow-up questions e.g. structural and contrasting questions.

I decided on the combined model to allow a more general discussion in the beginning that would give the participants a possibility to slowly get to know each other and at the same time provide an opportunity for them to take up issues they found important. More structure was applied later to make sure that the group discussed specific issues related to user participation, use of tools and techniques for knowledge integration, and the role of user representatives and IT-experts in the ERP implementation. The time schedule for the focus group sessions including the questions and exercises can be found in appendix B.

Despite recommendations not to spend more than two hours in a focus group session ((Morgan 1997) and (Halkier 2002)) I decided on four hours with a longer break half way. I had more questions and exercises than could be conducted in two hours and at the same time I knew it would be very difficult to get the participants to come twice. Based on my personal experience with consultants I found it reasonable to expect them to be able to concentrate and participate in longer sessions. When working at customer sites they often have to be on the ball for a whole day. In addition to the brake half way I decided on using different aids and exercises to drive away monotony and at the same time focus the discussions in different ways.

I documented the focus group sessions video filming them and recording the sound track on a Dictaphone. Notes on the whiteboard were photographed and papers used for exercises were collected. Afterward the sound track was transcribed and the photographs were incorporated in the text.

8.2 Data analysis, 2nd learning cycle

For analyzing the material I was using mind maps to ‘code’ the unstructured discussions; where ‘coding’ means a systematic way of understanding and keeping track of research data.

In the first iteration the mind maps were just used to record different discussion-themes going on in the discussion-at-large. Thus themes in the discussion were noted as a branch with an ID who initiated the branch. As people replied or elaborated on a theme (belonging to a branch) I made a sub-branch with the substance of the argument and an ID (consisting of who said it, and line in the transcript) identifying who contributed the sub-branch. As I was going back and forth trying to understand how the participants perceived ERP implementations I realized, that a couple of metaphors had been introduced and elaborated, this was especially clear from the material of the first focus group, but could also be found in the second focus group. Thus I decided to look carefully for metaphors and identified three.

After I had identified the metaphors I started to “read” the arguments used throughout the focus groups in a different perspective, and I realized that the ERP experts perception of the need for e.g. knowledge about the user organization was related to the metaphors, and also their arguments for using specific design artifacts during the implementation process was

related to the metaphors. The metaphors weren't necessary mutually excluding and some participants contributed to more than one metaphor.

"*The true effectiveness of metaphors is their almost paradoxical ability to point up dissimilarities and contrasts between two objects while simultaneously demonstrating that there are considerable similarities between the objects being compared*" ((Weaver 1967) in (Kendall and Kendall 1993), p 150).

There are four main functions for metaphors (Weaver 1967):

- Supplying concreteness or actualization of an abstract idea
- Clarifying the unknown
- Expressing the subjective
- Assisting thought

Looking into how the participants moved in and out of different metaphors, and this could to some extend be explained by referring to different implementations experienced working for different vendors or different client organizations (having different implementation objectives) throughout their ERP career. All the discussion as well as the 'coding' in mind maps was produced in Danish. However, as this thesis is in English I have translated one of the mind maps – as an example – included in Figure 8.1.

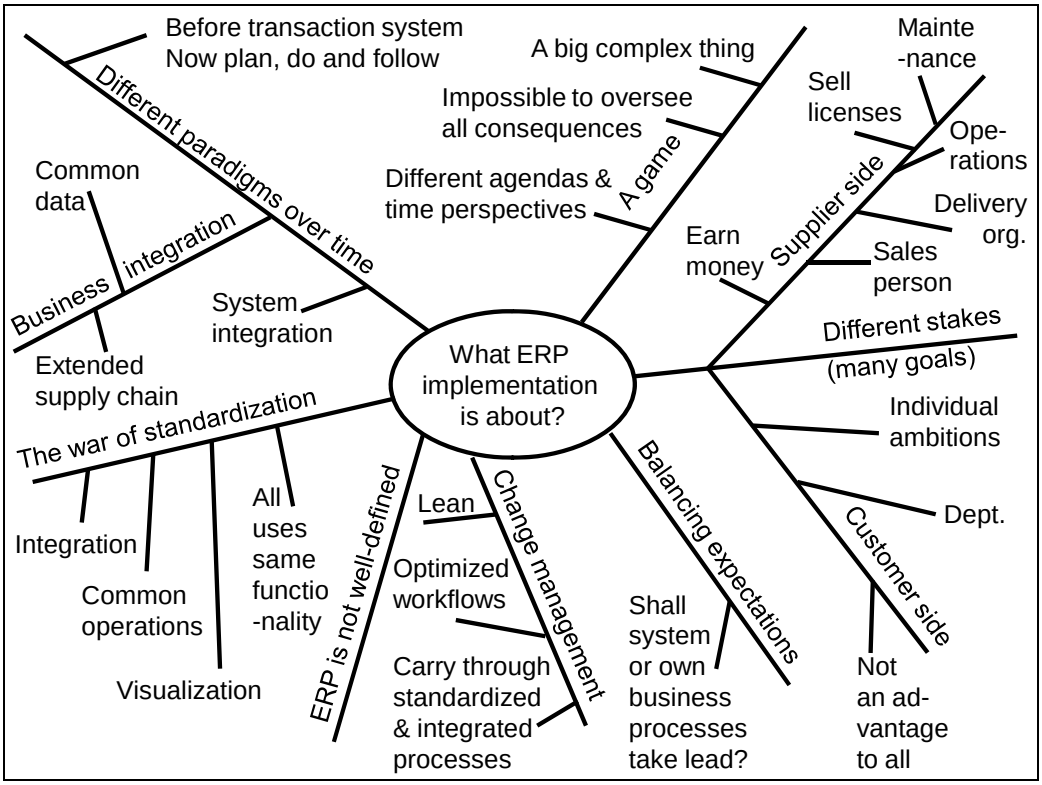


Figure 8.1. An example Mind Map from the coding of the focus group data

For the semi-structured discussions categories (very useful, somewhat useful, and not useful) is used to structure the analysis. Arguments in these categories are compared and contrasted in order to drive out the logic behind the arguments and the assumptions they are based on.

Chapter 9
Focus Group Analysis

In this chapter the results from two focus groups with external and internal ERP experts are presented. The analysis I present here focus on the ERP experts and their perception of an ERP implementation, how design is conducted, which techniques and tools are used for knowledge development and knowledge integration during the implementation, and the arguments for choosing them.

The focus group sessions had three main parts. One part investigation how the ERP experts perceived the need for ERP consultants to obtain knowledge about the implementing organization and their existing work practice. Another part investigating how the ERP experts perceived the need for organizational representatives to obtaining knowledge about the ERP software. And a third part investigating the ERP experts perception of what implementing ERP systems actually are about (the detailed research design and data analysis was explained in chapter 8).

Below I explain how this chapter is structured around these three main sections in the focus group study:

(1) ERP consultants obtaining knowledge about the organization

The first section presented investigating the perceived impotents of having the ERP experts obtain knowledge about the implementing organizations present work. I started with an open question related to my research question 2.b. In the concrete I formulated it: How important is it for the ERP experts to obtain knowledge about the specific organization? I have analyzed the answers given and report the results in section 9.1. After this open question I turned to a semi-structured approach. I asked the focus group participants to indicate, on a scale from 1 to 5, the usefulness of different techniques in relation to developing necessary knowledge about the organization. The result is shown at the start of section 9.2 and followed by my analysis of the answers of the semi-structured discussion that followed. In addition to the techniques included in the semi-structured question the participants added a number of techniques they found to be useful. I present these additional techniques in section 9.3.

(2) Organizational representatives obtaining knowledge about the ERP software

The second section investigating the perceived impotents of having the user representatives obtain knowledge about the ERP software. Again I started with an open question related to my research question 2.b. In the concrete I formulated it: How important is it for the user representatives to obtain knowledge about the ERP software? The result is shown in section 9.4. After this open question I again turned to a semi-structured approach. I asked the focus group participants to indicate, on a scale from 1 to 5, the usefulness of different techniques in relation to developing necessary knowledge about the ERP software. The result is shown at the start of section 9.5 and followed by my analysis of the answers of the semi-structured discussion that followed. In addition to the techniques included in the semi-structured

question the participants added a number of techniques they found to be useful. I present these additional techniques in section 9.6.

(3) The ERP experts' perception of what ERP implementations actually are about

Finally, in section 9.7 I present the analysis of the very first open question I asked the focus groups. In the concrete I formulated it: What are ERP implementations actually about if you had to explain it to someone not knowing about ERP implementations on beforehand? The analysis leads to the identification of three metaphors that describe the consultant perspective on ERP implementation.

9.1 Importance of ERP expert obtaining knowledge about the organization

In this section I initiated the discussion with the question: How important is it for the ERP consultant to obtain knowledge about the specific organization? My analysis lead to the conclusion that the ERP experts perceive it to be very important for an ERP implementation consultant to obtain concrete knowledge about the organization and the organizational practice, but they focus on understanding organizational processes on a more abstract level, and more on future processes (to-be) then on the existing processes (as-is). In this section I present my analysis and document it with numerous citations; arguing for and against the importance of knowing about the specific organization.

9.1.1 Why it is important for the ERP experts to know about the organization

First of all knowledge about the organization is important in order to interpret the requirements and understand the organizational background. One consultant talks about where the organization is coming from and which bindings the requirements have; meaning that requirements come with numerous assumptions and pre-conditions:

"…also that you need to know where the organization is coming from. Because it discloses the bindings (special requirements) *the organization believes it needs to have included in the new solution. Bindings needed, because it was something they had to live with; because it was deeply-rooted in their routine. But it does not have to be so if you take an outset in the new system. If you don't understand what is behind their wish to have something done in a specific way then you will be more prepared to accept e.g. that the VAT code has to be a part of the chart of accounts, even if it is not necessary in the new system…"*(Focus group 1 section 50).

The same consultant emphasize that knowledge about the current way of working as well as the current systems is of utmost importance:

"It is important to know how the old system work and the existing practice at a general level. Above all knowing why it is so. And then it is very important to question everything. As a consultant you should not accept, that in this organization it is impossible to do different. It is the biggest pitfall a consultant can fall into; not seriously challenging them and questioning them closely." (Focus group 1 section 50)

In general it is assumed that the requirements are derived from the existing practice but taken out of that context and made more abstract when the requirements specification is written. To understand the background for the requirements ERP consultants need to get in dialogue with users:

"It is important to include people from the user side who know the present organizational practice, those who know what will be best for the organization in the future, know the potential for improvements and where benefits can be realized. Know what it is that is costing too much; administrative costs, take too much time, where they are too late to get the order or ….It is important to understand the existing business processes at a somewhat abstract level. Above all understand why the business processes are as they are" (Focus group 1 section 50).

In order to discuss or even dismiss a specific requirement or explain how it can be achieved in a different way it is important for the consultant to understand the existing organizational practice. Thus the ERP consultants use knowledge about the organizational practice to argue why the standard solution in the ERP software is better than the existing practice or at least useful for the organization. This is said in the following way by one ERP consultant:

"… but the processes they need to set up they need to know about. The consultant also has to be able to show what the processes can and show its weaknesses. ….it is not enough to know the processes in the system you also have to be able to relate it to what other customers do, maybe explain difficulties they have using the processes." (Focus group 2 section 106)

Another ERP consultant supports this and emphasizes the interaction between ERP consultant and user, or business people as the users are called:
"It is important when you are in the process of designing the solution. In that phase it is especially important to understand what the solution should be able to achieve. You need to tell the user representatives:" if we do it like this then you don't need to do this". You need to understand their situation to be able to do that. Of cause you will never understand the business at the same level as the business people but you need to understand their processes on a general level. You cannot design solutions without that insight." (Focus group 2 section 100)

A different rationale for obtaining knowledge found in the discussion is that you need the knowledge to be able to find patterns in the organizational way of operating; patterns that can be related to other companies using the ERP software. When a pattern is recognized the consultant can draw on his experience to find a solution in the ERP software, as well as the arguments to convince (or try to convince) the user organization that they are not that special.

"I think it is about seeing patterns. When the users explain their business processes, then you need to recognize the patterns. They may use different concepts and explain it in a different way, but you can still relate it to the solution. It has to do with experience from other implementations…..parallels and of cause the system you use to implement them. " (focus group 1 section 51)

When seeing it from the perspective of the user organization it is important for the ERP consultant to know about the organization in order to understand and appreciate the unique characteristics of the organization:

"*I think you find organizations that are very special; with services that are very special. Therefore you also have to consider if you need a tailor made solution or you could actually live with a customized standard system."* (Focus group 2 section 150)

Thus concrete organizational knowledge is needed in order to understand why the standard solution is not working in the specific organization, and a customization may be needed. Part of this argument is that it is necessary in order to understand the work environment; the context the solution has to fit into. A good example of the uniqueness of a concrete situation was given by one of the participants in the second focus group:

"*I am very impressed with the way our system is adapted to the unique situation in our garages. First the shop foreman takes an order slip printed from the system with a barcode on, then he scan the barcode. He takes the slip with a special tool because he has oil all over his hands, and after scanning it he attaches the slip to his clothes. Then he goes to the rack with spare parts, takes the parts he needs and scans their barcodes. Then he does the work he has to do. After finishing he scans a barcode indicating the end of the job (automatically registering the time used). Everything is now updated; the time used, the spare parts used, the salary information, the stock on hand etc. I think it is rather important to know about the context when developing a solution, why the barcodes is important in this context, why you can't use a normal computer and the sequence of work tasks the standard process prescribe."* (Focus group 2 section 163)

Organizational knowledge is important in order to be able to map the organizational processes to the ERP-based solution. This of course is argued by the ERP consultants, but when seen from the organizational perspective it is emphasized that you take an outset in the organizational processes (organizational practice is guiding) not the ERP software when mapping.

"*I think it is very important. I have been part of the organization for a long time. You need to understand the organization that you are making the solution for. It is essential to the chances it will be a success at the end. It is essential to understand the processes and be able to map them to the solutions that are available in the standard system.*" (Focus group 1 section 46)

"*But still you have to understand the essence of the business, I think it is important that you understand that*." (Focus group 1 section 48)

If the consultants don't have sufficient knowledge about the specific organization then they will base their decisions on assumptions that may be wrong or distorted:

"*…..maybe it is more in the sense that they (the consultants) assume they know, and then solutions are based on assumptions. If some of the assumptions are wrong then the solution will be wrong." (Focus group 2 section 118)*

9.1.2 Why you need to focus on the products of the processes and on the future

Although knowledge about the existing organization is considered important seen from an ERP consultants perspective it is at the same time argued that the necessary and relevant knowledge may be at a more abstract level (high level); focusing on the final products produced by the organizational processes instead of the concrete tasks performed, the sequence of the tasks and the people performing the tasks.

"*If it's more a matter of how things are done, then you are too deep. It is more about why and what should be done. ……. If you get into how then you are in solution mode and that you are not interested in. Actually you do not care for the existing solution. It is much about what should be done and why*." (Focus group 1 section 54)

It is also argued that what are relevant may be a mix of understanding the present and the future. Business cases, project goals, improvement potential, and the concept of TO-BE processes are used as different ways to gain knowledge about the future.

"If I at all shall work with configuration, then I want to understand the business processes. I don't think a good consultant can do good work without an interest in looking into how it works. It is 25/75 on both side (consultant/ERP software and user side/business knowledge), it is not working without. Those that isolate themselves are not the once pulling it off. But it depends on the level, you said organization not processes, they are connected of course. If it is the architecture it could be about the organization, mega-processes or something (gets interrupted). ….. "When you say user organization, then I think it is not important. Organizations get changed in relation to which processes are important; the existing processes and the details in all corners." (Focus group 2 section 120)

"*It has to do with the objectives* …. One of the focus group participants gives an example of an automation of the quality control process of drug testing; going from 30 people in the department to 6 and from 25 days of process time to 2 days. The new solution would make the existing organization dispensable thus it was not necessary to involve the existing users or know the details of the existing process. "(Focus group 1 section 55)

Understanding the overall product and the business rationale – value if you like – behind the processes seems to be key:

"It is my opinion that you should understand it to some degree (the use domain). You need to understand the business, understand their customers, their products, their vendors, the level of the employees, and what is important and unique about the organization, in order to hang on to it. On a more general level you have to put distance to the existing processes. Not to say you can do without any knowledge about the existing business and the administrative functions. But you need to take an outset in the new system. For god's sake don't use to much time trying to understand and make as-is descriptions. But spend time understanding what kind of process you are trying to serve to the company. You should not listen to those who say the system should fit the organization instead of the other way around. For god's sake don't listen to them. You explain, and explain, and explain, and you need to understand where the gold is hidden, you need to understand what is important and hang on to it in relation to the project's objectives. You have to really focus on to-be. You have to scratch the surface of as-

is, but it is important not to spend months analyzing the existing processes." (Focus group 1 section 47)

The person stating the citation above (Focus group 1 section 47) places the responsibility for both the analysis and the design of the "TO-BE" processes on the ERP consultant, along with the responsibility to ensure that improvement potential is realized. Another perspective can be found in the citations below. Here the responsibility is placed on the organization. The organization should drive the project as a (business) change project. The potential for improvements and the way to reach the goals are defined by the organization. Relevant knowledge about the goals and the TO-BE processes are given to the ERP consultants. Thus someone in the organization is responsibility for the high level design of the future processes and communicating what is needed to the ERP consultants.

"I don't think you can place the responsibility on the ERP consultants. It is actually the organizations own responsibility. If you are not aware of that then you make a major mistake. … I think it is important to involve the very smart employees, and place them on key positions (in the project team) to enable them to ask the right questions to the ERP consultants, people who know the business, the markets, who know both the existing IT and the structure, the advantages and the disadvantages. People who know the organization 360 degrees and can see the connections between the ERP software and the business" (Focus group 2 sections 56)

Any communication process has at least two parts. Here we have the ERP consultants and someone from the organization. They need to work closely together as a team:

"I don't think it is so much about if you know the business (the consultant), but it is about having the right team. In the team both someone from the organization that knows the processes are needed but especially it is important to have someone from the vendor side, who know the system very well. The better team the more successful the project"(Focus group 1 section 49).

And the people constituting the team need to have extensive knowledge either on the organization or on the system.

" ……As (S) says, you need an experienced team, it is very important. On vendor side, knowing the new system very well and what has been done in different implementations, and experienced in quickly getting to know a new organization, their everyday life, their world, their life form. On customer side people that know how things are done now, people that understand what will be good for the organization in the future, where you can find potential benefits that could be picked up by this project; what are the unnecessary costs today, administrative costs; cost related to time issues, uncertainty or whatever resulting in lost sale or lost customers. Anything that can be important to improve; that the ERP project could be about. (Focus group 1 section 50)

One of the participants in the first focus group recognizes the importance of the team approaches described above but argue that the knowledge needed in the team might depend on the specific ERP consultant company and the approach they use to engage ERP implementation:.

"In my experience the suppliers of ERP implementation assistance can have two very different approaches. I have worked for two of the main suppliers. One of them focused on specifying the new system in cooperation with the customer and help implement it. The other had the idea, that the consultants should drive the project guided by business cases. Thus the mission was to define business cases and carry them through; realize them. In the first case you worked more or less for the IT department; you perceived yourself as IT consultants, in the second case you perceived yourself as managements right-hand." (Focus group 1 section 20)

As part of the discussions some arguments are also provided that it is unnecessary or very difficult/impossible to start out from AS-IS, meaning useful knowledge about the existing organizational practice. One set of arguments are based on the presumption that the system will change the way of working and operating because (it is presumed that) the system's implied processes are much more productive.

"…The organizations are changed anyway when you come with the new system; the processes depend on the system and the other way around." (Focus group 2 section 122) ….

"All the SAP templates require you to draw the processes and you are supposed to map them afterward to the system… But not many do." (Focus group 2 section 124)

Another argument is based on personal experience that the organizations haven't documented their processes and can't agree on them anyway.

"…. In my experience every time you start discussing and drawing all you get out of it… or the primary result is very long discussion, especially internally among the organizational representatives, about how they actually look (the others agree)." (Focus group 2 section 126)

In focus group 2 some of the discussion in the beginning focused on the need for "bridge builders"; especially the need for representatives from the user organization with extensive knowledge about the technology (ERP systems as such). This can be illustrated by the following citations.

"It is 25/75 on both side (consultant/ERP software and user side/business knowledge), it is not working without."

"In my department we operate with 25/75 profiles. 75% of the knowledge should be related to the business domain and about 25% in the area of IT. We have not fully achieved it jet….. and in our IT department it is the other way round. They need to understand some of our business domain. You focus on different domains but also have an intersection. That I believe is the key to success." (Focus group 2 section 77)

"But when you start the ERP project you have people who know the business 75% and the old systems 25%, and some IT people who know the systems that are now dismissed. The external consultants know the new ERP software 100% and nothing about your organization. You miss the intersection, which I agree you need. You need to build it over time." (Focus group 2 section)

The knowledge needed to become a' bridge builder' may come from having played the opposite role. I.e. being an ERP consultant before moving to an internal job in an organization:

"It is also about having former consultant employed in the organization, someone who is internal to the organization. We are able to ask the critical questions on behalf of the organization. We know the difficulties and the issued related to the software. If you don't have any internal people you will have to buy the skills externally." (Focus group 2 section 75)

But 'bridge building' is not just about knowledge. It is also about being able to influence the organization:

"It is about having a person with a leg in both camps. Someone who can tell the organization this requirement is unrealistic or too expensive to fulfill. It is a kind of interpreter." (Focus group 2 section 76)

9.2 The usefulness of different techniques

In relation to developing necessary knowledge about the organization the usefulness of different techniques were evaluated using a semi-structured techniques. First I asked the focus group participants to evaluate different techniques. For evaluation they used a scale from 1 to 5. The result is shown in the table below.

	1	**2**	**3**	**4**	**5**
	Not useful		Could be useful	Very useful	
Visits	XX			XXXXXX	X
Documents		X	XX	XXX	XXX
Requirements specification	X	X	XXX	X	XXX
AS-IS data model	XXX	XXX	X	XX	
AS-IS process descriptions	X	XX	XX	XX	XX
Use cases		XX	XX	XXXX	X
Discussions			X	X	XXXXXXX

Table 9.1. The summarized results from the focus group exercise where participants were asked to evaluate the usefulness of a number of techniques.

After the individual participants had scored the techniques I used the scoring as a basis for a discussion in the focus group. In the analysis below the argumentation for the usefulness of the different techniques are clustered in three different groups: First group; *not useful* contain grade 1 and 2, Second group; *could be useful in some situations* containing grade 3, finally the third group; *very useful* contain grade 4 and 5. For each of the techniques the arguments for placing the technique in one of the three groups are condensed.

9.2.1 Techniques discussion: Visiting the user organization

Very useful:
The majority of the participants find visits in the organization very useful. The general argument is that the consultants need to understand the specific organizational context in order to perform their work. The need to understand the organizational context however seems to be based on two different sets of arguments. One set of arguments relate to the importance of being able to communicate with the people in the organization, to get a background for understanding the requirements specification, and to be able to understand where the users are coming from in order to support the change process. Also the psychological effect showing your interest in the users and their work is emphasized. Thus the argumentation primary relate to the artifacts ability to address the *conception dimension* of design work.

Example 1: "*I think it is about going out to see the organizational culture. At a point in time the implementation consultant has to provide some kind of training. Also to see the users' IT maturity; their readiness in general.*" (focus group 1)

Example 2: "*You need to visit the user organization to understand the specific conditions for their work …* a *visit provides an 'aha' experience. You get a picture of reality that is what I think you gain from it. Personally I would appreciate the insight I got from such a visit."* (H - Focus group 1)

Example 3: "*It is also about the psychological effect. If you have taken the time to visit my organization I assume you understand and care for my organization whether you got something out of the visit or not*." (J – Focus group 1)

Example 4: "*One of the things you need to ensure in the project is lots and lots of communication to adapt and reconcile expectations. The psychological effect of a visit is considerable*."

Another set of arguments focus on the importance of developing knowledge about the final results/products produced by the organizational processes. This knowledge is important for the consultant in order to be able to design new work processes. The consultant sees his role as work process designer in the sprite of Business Process reengineering. Thus here the argumentation is related to the artifacts ability to address the construction dimension.

Not useful:
Only two of the focus group participants considered visits not useful. One argument was that the consultants need to focus on processes and get an overview of the processes, and in the

participants mind visits are not a good way to develop an overview of the process since many different organizational units is involved in processes.

Another argument was that it is not necessary for the consultant to talk to the end-users to get to know what they do: The consultants already know how to do what they do, and anyway the ERP-system provides one or more ways to do it, and the end users have to learn to do it that way.

Example 1: "*I agree, I don't need to visit the end users, I have no need to see what they do. I know what one have to do, and that is what the user organization has to learn. Because it is us* (the consultants*) who know how to do it best …. I mean how you best enter something in the system so that it is fast. Thus the process, the business process* (the organization's business process)*, that is what is important to learn not how users perform their work."*

Example 2: *"To see the existing system, that I don't think has anything to do with the system I have to help build. I mean, they of cause have to specify what they want to gain from the new system. Then I will help them put the building blocks together and then they need to decide afterwards if it is good enough. But knowing what they do today; knowing the existing system that has very little relevance to me. When you implement SAP you change the sequence of what you do and screens are very different, it is actually very different. There isn't much correlation."*

9.2.2 Techniques discussion: Documents used in the existing organization

Very useful:
The primary argument is that documents used in the daily operation are a very quick and easy way to gain insights into obscure details about the existing practice
Example 1: "*In addition it provides information about what external stakeholders has been used to. E.g. in relation to different discount arrangements. Often such issues are forgotten when discussing processes." (Focus group 1 section 73)*

Documents can also be used to develop an overview of the organizational processes.

Example 2: "*In my experience you can use much time discussing basic things but if you as an experienced consultant look at an invoice, export documents etc. then you very quickly get an idea about what needs to be included in the ERP solution." (Focus group 1 section 70)*

Documents catch the essence of the business as one consultant expresses it.

Example 3: "*It also a 5 for me, it is because they catches a lot of old habits and the essence of the business. E.g. if you have a document for complains it could initialize a process paying out a compensation at the end. Documents related to processes often uncover things you as an individual don't know. Documents can provide an understanding of the essence and the complexity of the business extremely fast." (Focus group 1 section 74)*

And finally documents may reveal things that are not in the requirements specification.

Example 4: *"In many cases documents provide information about things organizations wouldn't think about including in the requirements specification." (Focus group 1 section 75)*

Could be useful in some situations:
There are two sets of arguments. One is that documentation of existing organizational processes or procedures seldom are up-to-date/correct and therefore of limited value.

Example 1: "*I have it as a 3, but my focus has been on old process documentation and procedures and they are not useful. There is a difference between what is written and what is actually happening.*" (Focus group 2 – H)

The other argument recognize that external papers and other documents have to be specified at one point in time and that existing documents could be used as a kind of check list or inspiration, but they are not believed to be rely important for the design work. When listening to the arguments from those scoring it as 4 or 5 the reaction go in two directions. One reaction is to dismiss the argument because in their opinion the requirements specification should include what is important for the organization. The other reaction is agreeing with the importance of documents related to daily operations, but maintain the score 3 for documentation of processes and procedures.

Example 1: As a reply to the argument for the usefulness …… "*It should be specified somewhere in your process diagrams, data model, requirements specification … somewhere." (Focus group 2 – S)*

Example 2: "*I decided on 3, because documents can be many things. I can see that I agree with you that the final product; an invoice, a salary statement etc. can be good to take a short look at, but other kind of documents; old process descriptions and business procedures do not reflect the reality.*" (Focus group 2 – H)

Not useful:
Only one person found them not useful and also here the score is based on an evaluation of the validity of process documentation and descriptions of procedures.

Example 1: "*I am surprised that you all score documents that high. I haven't been on the vendor side that long but in my experience documents are very seldom up to date, and they do not reflect reality.*"

9.2.3 Techniques discussion: Requirements specification

Very useful:
Four people in focus group 2 find requirements specifications very useful (none in focus group 1 scored it higher than 3). The requirements specification is considered the most important document because it enables both parties (users and consultants) to understand what is needed (wanted) and what is delivered. It is pointed out that in the consultants' understanding a requirements specification is not a document developed solely by the user organization prior to the consultants entering the project. But it is a document user representatives and consultant develop together.

Example 1: "*I started out today saying that the most important is to develop a requirement specification, and to make the customer understand that what is specified is what they get. Maybe you think that I expect the customer to deliver the requirement specification, but that is not the case. In my eyes it is not the user organization that has to develop the requirements specification on their own. It is something you work on for a long time together, very long time. It goes back and forth to be adjusted and detailed …Because some things can be absurd and some things you don't understand. But both parties have to gain an understanding about what you want (need) and what you get*." (Focus group 2 – E)

It is strongly emphasized that the requirements specification is important in order to make the organization understand what they will get. The requirements specification is also seen as an important document as the basis for change request management. Thus it is especially the requirements specification's ability to serve as an explicit mean for cooperating that is valued.

Example 2: "*I don't think the requirements specification needs to be detailed, it should start the discussion. That is where I find it valuable. If the customer don't have an idea then it is difficult, but if the customer know at a more general level what should come out of solution. Then we start there and move on*." (Focus group 2 – S)

In focus group 1 this emergent character of the requirements specification is discussed and it is suggested that it may be more appropriate to consider it a description of the solution. There is a discussion and some agreement that the requirements specification also could reflect the solution, but it is still considered more important that the organizational (business) requirements are specified, and that the customer are the once deciding on the requirements. Some of the arguments seem contradictory e.g. on one hand the requirements specification is the customer's responsibility and could (should) be defined at an abstract level to allow a discussion and leave room for different solutions. On the other hand it is considered a document specifying exactly what the customer wants/gets.

Example 3: "*A requirements specification can also be a description of specific processes that you would like the system to meet including specifying, response time, working speed etc. That could be what a requirements specification is about. As I see it what is important is that the requirements specification is seen in relation to the test you should perform. Therefore you need some kind of relation. But I have only scored it as a 4 because there will always be things you couldn't specify or didn't know about and you find better ways to solve."* (Focus group 2 – H)

Example 4: *"So they (the user representatives) of cause need to specify what they want to get from the new system. Then I will help setting up the system."* (Focus group 2 – S (visits))

Could be useful in some situations and not useful:
Those who have given requirements specification 3 or less emphasize the difficulty passing on knowledge about the organization and the organizational context to the consultants via the requirement specification. The difficulties expressing on beforehand what you need is part of the argumentation, and from their perspective it becomes a serious issue because the requirements specification is expected to serve as an explicit mean to regulate the cooperation.

Example 1: *"It is difficult because when I make the requirements specification then I have committed myself. If you deliver something that can justify the requirements then I have to accept it. In some ways it seems as letter of marque. I admit that I sometimes miss something when making a requirements specification. …."* (Focus group 2 – J)

Example 2: *"How should we then know what to do?"* (Focus group 2 – E)

Example 3: *"Well, yes, but it is difficult, and that is why I think we could use some of the other techniques to allow the consultants to better understand the organization and its context. " (Focus group 2 – J) …… There is an information gap between the customers' anticipations and expectations, and the consultants' knowledge and understanding. How can we best uncover it as early as possible?"* (Focus group 2 – J)

9.2.4 Techniques discussion: AS-IS process data model

Very useful:
Only two of the participants consider as-is data model useful, and they have both marked it as 4. For them it is part of their work; mapping the existing data to the new data model although they admit that a formal data model is very seldom provided. In that case they have to be more creative to collect the information.

Example 1: *"It is far from all customers who provide a data model, but some professional organizations do. Sometimes the customer just provides some descriptions of the old system"*(Focus group 2 section 185).

Example 2: *"We make a mapping from the old to the new. I need to know what data you need and how they look if I should be able to set it up. ….. If you haven't specified it I will ask you"* (Focus group 2 section 187).

Not useful and useful in some situations:

For both categories the argument is that the customer very seldom provide a data model, and that an AS-IS data model only is used very late in the implementation normally for data conversion.

9.2.5 Techniques discussion: AS-IS process descriptions

AS-IS process descriptions are the technique with the most diverse scoring; using all five scores with a very even distribution. But the two focus groups are divided in this matter. Focus group 1 only has scores from 1-3 and focus group 2 has one score at 2 and the others are 4 and 5.

Very useful:
The argumentation for the importance of the AS-IS process descriptions go in many different directions. To some it is not the AS-IS process descriptions included in the blue print that is

important, but instead the process developing AS-IS descriptions together with the users and discussion them. It is emphasized that this often makes the organization realize that processes are too complicated. It is also argued that AS-IS process descriptions are important because you need to understand where you are coming from in order to change the processes. The process descriptions are also seen as a tool to make the scope of the implementation visible and to provide the basis for a gap analysis in relation to the ERP package software.

Example 1: "*In SAP we have some process descriptions of the processes implied by the system, the processes are in some areas problematic…. I don't see it as my task to change the processes but rather to find ways to support them. I find the process of drawing up the organizational AS-IS processes very important because it make the organization realize how complicated their processes actually are. It inflates that there are way too many exceptions."* (Focus group 2 – E)

Example 2: "*Obviously when you are changing process, you will have to take an outset in the existing processes*." (Focus group 2 – E)

Example 3: "*I also see that AS-IS processes has a relevance as a scoping device. It can be used to define which processes are included and which are not*." (Focus group 2 – L)

Example 4: "*To me the AS-IS processes is important because they provide an opportunity to realize the gap between the defined processes and the ERP system. Because what is important is to figure out where it becomes complicated to make the system meet the reality. We have to discover the challenges."* (Focus group 2 – J)

Useful in some situations:
The people giving it a 3 – useful in some situations - did not provide any specific argumentation in the focus group discussion.

Not useful:
The people considering AS-IS process descriptions not useful point to the lack of relevance for the future.

Example 1: *"I think you should use 20% of the time looking back and 80% looking forward. I don't think it is relevant knowledge for the consultants. …. In my organization we use AS-IS process diagrams to find the need for change. In relation to the consultants we would rather define goals in relation to the organization; business strategies and derived goals. Our AS-IS processes are not efficient or desirable. It doesn't add value to look backward."* (Focus group 2 – H)

And when it is not relevant it is a waste of time to use this techniques.

Example 2: "*Maybe I am colored by one of the biggest implementations I participated in. We had IBM develop such a blockbuster with AS-IS, but it did not reflect where the organization wanted to go. Thus they were not used for anything. …… process diagrams are able to pass on knowledge, but we are not really interested in AS-IS, what is important is TO-BO …… People like to tell the story about how they do things but it is not important we are moving forward."* (Focus group 2 – H)

9.2.6 Techniques discussion: Use cases

Very useful:
Although the majority has score use cases as 4 or 5 no one is really enthusiasm in the arguments for using them. One argument is that it is a tool that can be used in some situations e.g. if the user representatives in the project know too little about what is going on in the organization. Then they can be used to include more people/opinions in the process in order to illuminate a specific area. Another argument is that they can be used to balance expectations in the user organization.

Example 1: "*They can be used as examples or to validate a specific understanding; how things in reality are related. Often people participating from the organization know too little about how the organization actually operates. Use cases can be one way to illuminate what is actually going on."* (Focus group 1 section 79)

Example 2: *"I have used them internally in the organization to balance the expectations. In order to minimize the amount of new wishes coming up during the process. It can be a good idea to develop a number of use cases where all the ideas can be drained of. Afterward a number of use cases are chosen to be implemented. Then you have reached a reasonable scope."* (Focus group 1 section 80)

Not useful:
There is no strong argument against it, but people rating in low do not have any personal experience using it.

Example 3: "*Some of my projects have used them a little in situations where they found it necessary to agree on how to understand the process. In principal it has not been important to use, and it has not been used much. It takes much time to develop and it seldom adds much value. It is an ok technique, but it is not important." (Focus group1 section 82)*

9.2.7 Techniques discussion: (ERP) discussions

The consultants consider (ERP) discussions (both in a workshop like environment and just sitting together) the primary technique to obtain knowledge about the user organization, a large majority scores it as a 5. The two more skeptical participants are the two that have never been external consultants during an ERP implementation, and as user representatives they feel a little tricked in the process. Investigating the arguments closer it is clear that the focus group participants have difficulty separating the techniques ability to provide the consultants knowledge about the organization and provide the user representatives knowledge about the technical options. (ERP) discussions seem to be at the core of the design work; having knowledge from the user domain and technical domain (ERP system) present at the same time and integrated in order to decide on potential solutions.

Due to the fact that only one person in focus group 2 has rated (ERP) *discussions* as a 3 the discussion of the technique take on the character of finding out why he is more negative than

the rest of the participants. Thus the discussion only provides indirect arguments for the value of the technique. Below I will try to extract the main arguments for and against the technique, and instead of examples in the form of citations from one person I will provide a sequence of the discussion in focus group 2 to illustrate the dynamics of the discussion as well as the main points.

The first part of the discussion show how the value of discussions is challenged by claiming that discussions never end.

Opponent: *"Sometimes the result is endless discussions. The meeting was supposed to end at 2 pm but at 7 pm we are still discussing why I need to use the standard solution."*

Defending (E): *"And in your opinion you don't need it or you can't be bothered or what???"*

Opponent: "*It is often the case in discussions. We are still there hours after we should have finished. Discussing why the standard solution can/can't be used."*

Defending (S): *"And you don't think you need the discussion?"*

Opponent: "*In my opinion at some point in time you need to tryout other techniques. Stop the discussion and e.g. visit the department.*

The main argument for finding (ERP) discussions less useful is the feeling that the ERP consultants only care for the standard solution and that the (ERP) discussions take more character of lecturing the user representatives why the ERP software's inbuilt processes are better or at least useful for the organization. In case of disagreements (the user representatives reject the suggested solutions) then the user representatives get the impression that the ERP consultants don't care for the organization and its special circumstances.

Defending (S): *"... I can simply not understand how the consultant can do the design, based on what…. just looking at what is happening. It doesn't work for me. Requirements you don't want to specify, a data model is not provided, process descriptions may not be useful because the reality is different, what do you want me to use?"*

To ease such a situation some of the other techniques are suggested as more appropriate in order to provide conception regarding the organizational needs.

Defending (S): "*But what is your standpoint, you don't need the discussions? When you have chosen to use a standard solution then you need to design it. Can people (the consultants) design it without discussing it with you and your business people (user representatives)? Can it be done without your accepting to take part in it?"*

Opponent: "*If I am told that I can only have 8 out of 10 requirements fulfilled then I need to go back to my organization and discuss with them if we want to make customizations, give op IT support for specific tasks or if we should find a different ERP system. But long discussions with the consultants will not provide the answer.*"

Defending (S): "*No but the discussion could just be about finding opportunities or alternatives. All solutions have many options you can say. It is you (the business people) and maybe the architects who have to come to a decision. When you have many options then I find discussions very useful.*"

Opponent: "*Of cause we need the best solution*

There is also a feeling that the user representatives some times are taken prisoner in the process. The user representatives are professionals within their own domain, but not necessary strong enough to go up against an experienced ERP consultant in a discussion. Thus the user representatives may sometimes accept solutions they do not agree with or do not understand.

Opponent: "*I have seen consultants taking my users hostage. I know I may be pushing it to the edge but sometimes the user representatives are not used to discuss with consultants. They are domain specialists and they know their domain well, but they are not always good at standing up to consultants who tell them that an important requirement is obsolete. I therefore find it important to emphasize other techniques.*"

The last part of the discussion is actually illustrating the opponents point. Especially the participants (E) and (S) are absolutely certain that any company can use the SAP standard and if you cannot convince the users in a discussion you just approach top management who will tell the users to adapt to standard. No need to be sympathetic with the user organization or to care too much about understanding the organizations way of working.

Opponent: "*This is where I say it is not good enough; I can't live with one of the 500 standards."*

Defending (E): "*You can live with one of the 500 standards because 30.000.000 companies all over the world do it. They can apparently live with it!"*

Defending (H): "*It depends on knowledge, that the consultants have knowledge about the user organization. I just think you get it through the discussions, directly and unfiltered hopefully. I think it is much more important than all the documents.*"

Opponent: "*I agree that you get something different through the discussion. Of course you do, but sometimes I experience they end in a conflict.*"

Defending (E): "There is no reason for a conflict, you just leave each other and then you go directly to top management, and then they tell you to use the SAP standard!"

Participant (H) is however providing a slightly different argumentation why discussions are important. He indicates that the user organization is not necessary the best to realize the ineffectiveness and inefficiency of the existing work processes, but since you need to include someone who knows about the organizational practice you are in a dilemma. Therefore someone **professional** also needs to be included.

Defending (H): "*I think much of the irrational behavior in the different departments ……..but you cannot take all the users out of the process, I assume someone knowing about the user domain has to participate. But someone also needs to be professional.*"

Whether he means ERP consultants and/or internal process consultants is unclear. (H) Agree with the opponent that knowledge about the organizational practice is important for the ERP consultants, but in his opinion that is exactly what the discussions are providing - and much better than documents. He presumes that the discussions take place in a different spirit; that the participants are open to each other and truly are interested in learning about each other and the respective domains.

Defending (H): "*I assume it is a productive discussion, that we explore and challenge each other's knowledge and understanding. It should be fruitful; provide an understanding who are you and who am I and result in a belief that we can cooperate about this. If you approach it that way then I believe you understand some of what I am telling you, and you believe I understand some of what you are telling me*".

9.3 Additional techniques used

Besides the techniques discussed above based on my semi-structured interview guide for the focus group interviews a number of other techniques were brought up in the discussion; especially in focus group.

9.3.1 Additional techniques: Workshop based on demo data

The suggestion is to perform a workshop showing the standard system based on demo data. Basically it is about showing the organization how the system would work for an organization as the consultant understands the specific organization. In the argumentation for the technique interdependence in the knowledge development can be found. It is a way to show the user organization they are not unique, and make them realize that the standard solution may work for them. At the same time it allows the consultants to develop knowledge about the areas where the organization believes it differ from the standard.

"*A workshop based on demo data, showing a standard organization. I did it much in my time implementing Oracle. In my experience it is a very quick way to get a dialog with the organization. It takes them away from their humdrum everyday life. It makes them realize they look like a lot of outer organizations and that there may be best practice standard way of working that you can start to use. (My reply: don't you use it for the opposite namely providing the organizational representatives knowledge?) It is to start a dialog can we use it or not; to adjust expectations."* (Focus group 1 section 84 + 86)

"I agree with you, especially as it is common for all organizations in Denmark to believe they are unique. It is a very good tool to show them that they are not unique. It is also a good way to put some ideas to work, and start the process towards making them understand that they <u>can live with</u> a standard system. Many organizations believe it will never be possible for them. "(Focus group 1 section 87)

"I also have good experiences showing the system to the organization in a way reflecting the way I understand the organization, a gap analysis. You show the system and every time the organization claims they can't live with it you make a note. You show the system to the organization and the organization make a gap analysis, documenting all the areas what we showed them wouldn't work. That is providing you (as a consultant) a lot of knowledge about the organization. "(Focus group 1 section 88)

9.3.2 Additional techniques: Rich Picture

"*In theory this technique is very close to Rich Pictures that some of you may know. Basically it is about gathering people around a whiteboard and try to visualize the processes. Through that I get knowledge about the stakeholders and the processes that are actually performed. This is an alternative to use documents; you strive to draw the processes at a very general (abstract) level on the whiteboard. This technique opens a completely different dialog. It opens up a dialog about things that has never been documented. The unwritten is exposed*." (Focus group 1 section 90)

"*I agree people who know how to use a Rich picture or a similar technique they have a very good tool*." (Focus group 1 section 91)

"*I have tried the technique many times and it is possible to make more people contribute."* (Focus group 1 section 92)

9.3.3 Additional techniques: Externally provided TO-BE processes

"*I am use to work in a way where you make TO-BE processes on a high level and then you start a gap analysis. But sometimes you can also find external sources that provide TO-BE processes. E.g. it could be a governmental association that provides it. It could also be related to EDI standards."* (Focus group 1 section 94)

9.3.4 Additional techniques: Demo of the existing IT system(S)

"A demo of the existing system … It can be a good way to get to know the existing practice; to understand the organization. It can be a way to open your eyes for the organization." (Focus group 1 section 96+98)

"It is often a bad experience, in many cases you know that what you are providing may be 10 times less to the point (useful) then what they had." (Focus group 1 section 99)

"It is also my experience; when you implement a new system many of the very good functionality the organization had is diapering." (Focus group 1 section 100)

"It is true, but there is a very good argument; you get the big coherent integrated solution. One that doesn't require the same kind of ….(they speak all at once) … You really have to be

prepared for that, and you need to talk much about it. In all areas you will only get an 80% solution. But then you also get 80% integration. " (Focus group 1 section 101)

9.3.5 Additional techniques: Clarify structure and ownership

"*I have added that organizational structures are a way to understand the organization. It could be the legal structure or the management structure. Especially the legal structure, thus I have experienced that the people from the organization I worked with didn't know it at all. "Do we have three holding companies????" You need to know it as a consultant when implementing SAP. "Maybe in practice it is another company that has the legal ownership to the goods in the warehouse*." (Focus group 1 section 102)

9.3.6 Additional techniques: Look at legislation and salary agreements

"*I have also added regulations and salary agreements etc. Much understanding can be embedded here.* "(Focus group 1 section 103)

9.3.7 Additional techniques: Management interview

"*Management interview is also a good way to get a helicopter perspective. It gives me a more tactical and strategic perspective on what is happening in the organization. Most people form management is good at it; telling e.g. why a specific department or factory exits. You can talk much about what is happening in a department or factory but why does it exist? Maybe it is for tax reasons, or maybe you are only allowed to operate (sell your products) in the specific country if you have a production in the country. Maybe the factory pollutes or is not profitable, but have to keep it. This is also an example why it is not always good to keep challenging their actions but instead strive to understand the reasons behind. You can ask management about many areas; why do you use these suppliers etc. You get the big picture when interviewing management."* (Focus group 1 section 103)

9.3.8 Additional techniques: Interview external stakeholders

"*Taking to the customers; the implementing organizations customers. This is of cause more important if it is integrated processes. In my organization we had to develop an online shop for our trade customers. We did talk to the customers and we did understand that most customer organizations wouldn't use the new system themselves because they had outsourced this area, and that part we did get right. But what we didn't understand was that many organizations used externally employed people to perform part of the process, and this people would need access to the new system. External people never before had access to internal IT systems in the organizations. That was a barrier we didn't realize until going live, now it is solved. This is an example of a situation where we didn't talk enough with the customer's customers."* (Focus group 1 section 104)

9.4 How much do the users need to know about the ERP software

We shall now look at the second section of the focus group, this section is focusing on how much knowledge the user representatives should develop about the ERP software, and how they are expected to do it. I actually started out with the open question: How much do the user representatives need to know?

The focus group participants who use the consultant perspective have a very firm understanding of what the organizational participants in the ERP project have to contribute; they need to contribute requirements specification, test case development, perform the tests, and perform end user training. It is argued that especially performing the tests and end user training requires comprehensive knowledge about the system (as a user), but since these tasks are performed at the end of the project it is possible to build the necessary knowledge in cooperation with the ERP experts during the project. Using this perspective it is the final solution not the technological options that is important for the organizational participants to understand.

"*I think they need to contribute three main things: (1) requirements how the systems should work in this specific organization, (2) They need to test the system, and (3) they need to perform end user training. Especially the last two activities require of cause that you know the system very well. You cannot test it without knowing the system well and at the same time knowing the organization well. There can of cause be differences how you go about teaching but in general it is business representatives that within their area they need to understand the systems way of working very well.*" (Focus group 1 section 126)

It is also argued that those directly involved in the project they need to understand the implementation method and the systems philosophy and constructs, they need to understand the processes implied by the ERP system and they also need to understand what is easy to change and what is not.

"I think they (representatives from the organization) need to understand quite a lot. They need to understand the implementation method, that doesn't differ much depending on the ERP vendor I think. They need to understand the method you work with. They should also understand the system's philosophy and all the concepts. I don't think you should translate them to the customers "language", not anymore. I have participated in project in the past where we spend a lot of time translating everything to the customers constructs. We used a lot of time doing so and when you started using the system you had to use (learn) the systems constructs anyway. Another thing that is important is that the user representatives understand how the system works and what is easy to change and what is not." (Focus group 1 section 114)

But at the same time it is argued that it is a delicate balance. First of all you have to be aware that in many cases the user representatives doesn't have an interest or a talent in systems design, but it is also a delicate balance because too much knowledge may result in a desire to act as an "application consultants" wanting to design/change the application on their own. This is perceived as a problem because in order to do a good job as an application consultant (the one designing/changing the ERP software) you need to understand the system on the system's premises, and make design/changes with a long term perspective considering the

ability to upgrade in the future. Thus it is ideal if the user representatives get to understand the system as an advanced super user.

"*Sometimes I have experienced that at the end they would rather do it themselves … take over our work; the ERP consultants role. They want to design them self …. You can also get so deep into the system that you want to take over this part. Because now the user representative thinks he knows the system and there is no need for an extra link in the chain. Sometimes it may be ok, but sometimes you need to understand the system on the systems premises.* ….[]but they need to know the basic processes I think" (Focus group 1 section 117)

"It can easily become too technical. If you expect too much of the user representatives then it will go very wrong …..[] if you talk data types and codes internal to the system." (Focus group 1 section 116)

"*At the end of an implementation extensive end user training is of cause needed, but during the set up…I don't think they need much knowledge about the ERP system.* ***I think it is much more important that the application consultants understand the business*** *than the business people understand the system. I will articulate it this way,* ***I don't think they are interested or it is their skill****……...[]In order to do their part of the job, I don't think they need to know a lot. They will slowly develop a feeling for what SAP can. I don't meet deep knowledge or an interest in developing knowledge about what SAP is cable of. They will approach the application consultants to get to know it. They need some understanding but it is at the user level.* ***But what is important is that they have business knowledge and that they have the competence to change the processes, and know who to go to to get approval.*** *Processes are not something you just change. You need organizational knowledge and contacts to stakeholders in the process. But they only need limited knowledge about the ERP system. Many of the very good business consultants I have meet are not really technologically skilled, but they have a minimum of understanding for how the ERP system is and what is possible.*" (Focus group 2 section 265)

The last citation reveals that some ERP consultants have additional expectations to the organizational representatives. They expect them to have business knowledge of cause but also a network in the organization and contacts to decision makers, and they should be willing to promote the solution internally; become ambassadors for the project. Also in the other focus group such expectations were expressed.

"*They are also guarantees that the solution will work I practice; that it is the right solution for the organization."* (Focus group 1 section 127)

"*The primary reasons for involving user representatives could be internal sale, testing and training.*" (Focus group 1 section 128)

"*Then it is important they have decision power or can respond quickly. That they can take the decisions needed and stand surety for that it is ok*." (Focus group 1 section 129)

"*It could also be decisions that are difficult, therefore if they don't have contact to decision makers; process owners or department managers. If they do not have weekly contact then it is difficult for them to drive it forward. A classical mistake is to select someone from the IT*

department to participate; someone who don't know anyone in the organization who has to use it, it will not work." (Focus group 1 section 130)

"They need to be ambassadors and that they cannot be without knowing the system." (Focus group 1 section 140)

In the discussions the issue of involving different type of user representatives came up and different labels are used; part time resources, 100% resources, business consultants, internal consultant, super users and end users. End users and part time participants are seen as a problem.

"User representatives who has to look after their normal job on the side …..[] Them I would rather be without. You should just show them on a black board or in work-shops. It doesn't matter how much they try to be involved they are actually putting you to a lot of trouble because all they can cope with is concentrating on getting to know the system." (Focus group 1 section 115)

The labels 100% resources, business consultants and internal consultants are more or less used as the same and it is argued that they are the once who need the comprehensive knowledge about the final solution. It is also argued that 100% resources are preferable and that personal skills and prior experience with IT implementations may be more important than knowledge about the ERP system in order to successfully cooperate with the ERP application consultants. What is important is experience from prior IT implementations, willingness to learn and understand how to compromise.

"*The best business consultants are those who are willing to compromise, which understand reality and use standard processes. They are also the once who can specify what they need taking an outset in the standards the system provides. The business consultants who are sharp are those who have been involved in other IT implementations. They don't necessary need prior knowledge about SAP; what is important is that they are willing to learn; that they are open to look at how it could look different from what they are use to. Without that competence I don't think you should participate*."(Focus group 2 section 259)

"I also think business consultants need to know more than end users; they need to know a little more than they have to teach the end users. They need to be able to answer many questions. I have seen business consultants make small folders with tips and tricks in the system. This is fantastic; they can go to their colleges and say look, I may be able to show you something you didn't get in the training program. It just boosts the colleges trust in them." (Focus group 2 section 261)

Finally, it is difficult to communicate when not having a shared vocabulary

"It is very difficult to communicate with business people. There is a big risk that you will make a wrong solution because they have described what they need in a different terminology; one I am not use to. Thus I will have to show them how it could be if I have the possibility to do so." (Focus group 2 section 259)

"I think you need to teach them the concepts used by the system, sometimes you just don't understand what they are talking about. But as soon as you share concepts then it different." (Focus group 2 section 266)

9.5 Techniques for developing relevant knowledge

After discussing to what extend the user representatives need to know about the ERP package software the participants in the focus groups had to evaluate different techniques for developing the relevant knowledge. The result is shown in table 9.2.

	1	2	3	4	5
	Not useful		Could be useful	Very useful	
Standard education			XX	XXXXXX	X
Explore the system them self	X	XXXXXX	XX		
Walk through process models		XX	XXX	XXX	X
Sitting side by side with the consultant exploring screens	X		XXX	XXXX	X
Discussions with the consultant			XXX	XX	XXXX

Table 9.2: The summarized results from exercise 2 for both focus groups.

In the analysis below the argumentation for the usefulness of the different techniques are clustered in three different groups: First group; *not useful* contain grade 1 and 2, Second group; *could be useful in some situations* containing grade 3, finally the third group; *very useful* contain grade 4 and 5. For each of the techniques the arguments for placing the technique in one of the three groups are condensed

9.5.1 Standard education

The focus group participants agree on the usefulness of standard education. Thus 7 out of 9 have rated it very useful and 2 somewhat useful. As the discussion of the introductory question showed all the focus group participants agree that a minimum of understanding of the ERP system is necessary to work together with the ERP experts. Thus user

participants/business consultants need to understand the concepts and the basis processes in the new ERP system. The user representatives are expected to accumulate knowledge about the system over time, but need something to start out with "*at some point in time they need basic training*". Standard training is perceived to be the best way to provide a foundation; introducing the user participant to the concepts and processes within a specific functional area.

9.5.2 User representatives exploring the system on their own

The focus group participants also agree on "exploring the system on their own" but here the conclusion is that this is not useful. Thus 7 out of 9 have scored it not useful and 2 somewhat useful. The main argument is the complexity of the system. It is perceived as absolutely impossible for an inexperienced user to set up data and figure out how processes are performed using different screens. Having users try to explore the systems possibilities on their own is expected to result in disappointment and frustration.

Useful in some situations:

One of the external ERP experts who work with a smaller ERP system find it very common for users to explore the system on their own.

"I think the difference is that I work with a smaller ERP product, it is easier for the users to explore the system on their own." (Focus group 2, section 288)

Some of the ERP experts point out, that in some situations it might be ok if the user representative has a case they want to tryout. But in general the system is too complex for the user representatives to make sense of it on their own.

"I have formulated it a bit different. I think it is ok if the user representative work with a test system and have e.g. a use case….[] but if they just go in and try this or that then they won't benefit from it, they won't get far…..[]If they don't get any guidance they will get lost" (Focus group 1 section 166)

Not useful:

One of the ERP experts stresses that it is only very late in the implementation process it will be possible for the users to work with the system. It needs to be configured and master data need to be entered.

"Unless the implementation process is almost completed you won't get far" (Focus group 1 section 167)

9.5.3 Walk through process models

There seems to be some disagreement if and how to use process models. This time there is no pattern related to if you are internal or external ERP expert, or which ERP product you work

with. But the discussion gives some indication that the ERP experts' personal competences may influence their attitude. "*If you are able to show the end-to-end process it is cool ….. [] as I see it, it is often a real challenge for the consultants; they don't understand the processes across 3-4 modules. Personally I often have been in trouble. I don't know all the modules that well*." (Focus group 1 section 161)

Very useful:

Those who see process models or process diagrams as essential emphasize using end-to-.end processes to facilitate insights how the processes work in other departments, and that they then can be used to argue why one department might only get an "80%" solution. "*I think it is a good way to provide an overview …[]and the systems philosophy ….[] It also show what happens in other departments. Thus it can be used to defend why you cannot get a 100% solution, but have to live with an 80% solution*." (Focus group 1, section 147)

It is also emphasized that if process models becomes concrete, not generic, then they are very useful. If they get detailed enough so that it is possible to link them to concrete modules and transactions in the ERP software. "*….if they become concrete process models, not generic. So that you afterword can link them to, and that you can in SAP, link them to concrete modules and transactions, which should be executed, that way they are a good way to understand the relations"* (Focus group 1 section 157), and he continue explaining that if it is this kind of process diagrams then they can become an important means to illustrate why someone may have to do extra work in order to help someone else ware in the process flow. " *…also help understand that if you have to do something which is of no importance to you* ..[] *if you don't do it then someone else cannot do what they need to do"*. Thus they can be used to illustrate relations or dependencies between different actors in the organization, who might not be aware that they actually depend on each other.

For some of the ERP experts the process approach is vital to implementing the ERP software "*Both understanding the models build into the system and understanding the processes defined for the organization. The process tool is the most important."*(Focus group 1 section 153)

One of the ERP experts also reflect on one of the new trends he has experienced, namely that SAP now release new processes instead of new modules or functions. "*I have noticed that SAP now is marketing and releasing the system very process oriented, every time something new is released then it is a new business process, not a module or something else, it is a business process"*. (Focus group 1 section 155)

Useful in some situations and not useful:

Those who are more reluctant emphasis that process models are too generic, that they might work for any ERP product. "*at one point in time Oracle hired some professional process consultants, they could perform fantastic process demos but in reality the processes could match any ERP product, they never reached a level where the rubber meets the dirt"* .

In the second focus group it becomes clear that the SAP consultants make a distinction between the process flows provided by SAP and the process flow that is developed for a

specific client. The process flows provided by SAP are perceived to be at a very abstract level, meant for managers who only understand the processes on the conceptual level. "*What comes with SAP is main processes ….[] they are at the top management level*." (Focus group 2 section 315). Three of the SAP consultants are very puzzled why I am interested in process models and especially those provided by SAP, they cannot remember ever seeing anything from SAP at a detailed level "*We didn't get anything at a detailed level like work flows when I was a SAP consultant, actually the consultant company I worked for they made their own templates with their own standard processes which we sold to the customer*" (Focus group 2 section 332).

It becomes clear that the normal procedure for the SAP consultants is to ask the customers to describe their processes (mix of as-is and to-be) and then the consultant will transform the description to a process diagram, not the other way around. "*Normally the customer will provide it, and then you draw a process.*" (Focus group 2 section 335) But after some discussion it is agreed that the process diagrams developed (in cooperation with the user representatives) are used as a common reference.

9.5.4 Sitting together with the consultants exploring the system

Most of the participants find it very useful to sit next to the user representatives showing the possible solution and discussion them. Only one participant is very negative.

Very useful and useful in some situations:

One positive argument for this technique is that it is used at the final stage agreeing on how the solution will be implemented. *"It may sound strange, looking at screens, but at one point in time you have to agree this is the process and then get it documented and signed off….. [] showing what it is we actually agreed on is necessary, therefore you show the solution using the system …[]going in there doing this, then going in here …."* (Focus group 2 section 359).

Another argument is that it is a way to build trust; trust in the consultant and trust in the solution, seeing the screens is very different from written documentation or discussions *"It is about confidence; the user representatives develop confidence in the consultant when being close and seeing the screens. Asking user representatives to read documents doesn't work in my organization. Therefore, actually it is important to see screens."* (Focus group 2 section 360)

Not useful:

"*I think it is a controlled and seductive process. The user representative sit next to the consultant who control the computer and say see you do this, and then you do this ….[]the user representatives are taken prisoner, they don't really understand what they accept.* (Focus group 2 section 361) The other ERP experts, especially the external are astonished and a bit puzzled; obliviously the critique is surprising to them. "*It is a dialogue; if the user representatives don't understand then you need to make drawings, discuss, explain…show it on the screens.*" (Focus group 2 section 368) "*Maybe the consultant is in control but it is a dialogue*" (Focus group 2 section 369). But it is worth noticing that it is the same consultants

that less than 30 minutes ago tooled, that if the discussion between the user representative and the consultants could not be resolved then they relied on management to demand the user representative to accept the consultant's solution; the standard solution (section 9.2.7)

9.5.5 Discussing with the consultants

The score is with a single exception the same for sitting next to the consultant and discussions with the consultant, and at this point the discussion in the focus group the two techniques get mixed up. Going through the transcripts no new arguments are provided. In section 9.2.7 arguments why discussions involving the ERP expert and the user representatives is important is provided.

9.6 Additional techniques suggested

In focus group 2 they didn't have any suggestions for new techniques at the end of the evaluation but during the discussions a little later a few suggestion came up. In focus group 1 a number of suggestions were put forward.

9.6.1 Internal demo before and during implementation

"Some customers have much focus on internal acceptance, I have once tried to set up simple scenarios based on the existing processes in the old system and show it to the organization.,,,[]but I takes time, and if you are on a strict time schedule then internal acceptance have low priority." (Focus group 2 section 273)

"*I have tried to use a method where we accelerated the process; accelerated the understanding, the acceptance, education and the change process. What happened was that a number of end user demos were performed during the ERP implementation based on scenarios. SAP's implementation methodology and ASAP is very rigid, you could say that what we did was to split both blue-print and the implementation phase in several prototype séances*." (Focus group 2 section 275)

9.6.2 Workshops, prototypes

It could also be understood as a work-shop performed for process owners and end users not actively participating in the ERP project work. In that case the advantage is to allow a larger group of people to get familiar with the system. (Focus group 1 section 169)

"I come to think about that in general it is not necessary to perform prototypes/demos because most of the processes is boring administrative processes. It is not innovative processes." (Focus group 1 section 171)

"I think there is some prototyping in sitting together and design based on an idea or discussion you had with a user representative. Sometimes you need to tryout different

possibilities. In my experience that is often very successful; when the user representative gets some screens to respond to, then they can give useful feedback. It is very different from something abstract on a wall or piece of paper." (Focus group 1 section 172)

"*I agree, but in general this doesn't take place until after go-live. Where you e.g. because of performance difficulties have to consider how to optimize the process….[]but if you start out using it as a part of the ERP implementation, then it will draw out on forever.* (Focus group 1 section 173)

"*..but it could be a controlled process…[]important issues you have to consider together …[]I think it is a very useful."* (focus group 1 section 174)

"*Prototypes take a long time to establish especially on live data…. []if it is a standard process then you don't need a prototype, but if it is something more complex you need to explore then you might need a prototype. However it takes time."* (focus group 1 section 181)

9.6.3 Early training and training in the new processes

"We realized that statistically we used 45 hours on average per employee when implementing in a new location. In the first implementation much of the time was spend after go-live. We realized that it was much more effectively using the hours before go-live. The statistic showed that the first 2 weeks after go-live the productivity was only 60% and it took 4 month to get back to 100%. But changing the timing of the training we were able to get back to 80% in three days. (Focus group 2 section 279)

9.6.4 Collaboration lab

"*We have tried a new type of workshops, 2 times 12 hours workshops. It is a very special experience. We had many whiteboards that was moved around, music,,, [] The idea is to bring the right people into play, the stakeholders; end users, super users, managers, project management, steering committee, external ERP experts and other consultants. Then we had different themes (processes) e.g the process AS-IS should be documented, we had a suggestion on the whiteboard and then people had to go visit the different places. This was a very effective process. …[]the point is that in 2 days you can perform the work that it normally will take many weeks to do because you never can reach the people you need to ask something….[]we bring the stakeholders together in the same room. The quality of our project increased dramatically….[] in our organization many people normally go home when they spend the required 7 hours and 24 minutes at work, but these two days they gladly staid*." (Focus group 2 section 378+380)

"*As a response to this one of the external ERP experts respond [] "I think it is more important to get the dataflow in place across the functional areas, a data model. And it that regard I think the ERP experts are the more important actors."* (Focus group 2 section 386)

9.6.5 Best practice accelerators

I once worked for a consultant company who had developed a database with SAP processes. Actually it was sequences of screens recorded that could be played again and again for inspiration. It was kind of a demo of all processes. The customers loved it. It wasn't something SAP provided it was our own. (Focus group 1 section 191)

9.7 Three metaphors reflecting the consultants perception of ERP implementations

As explained in chapter 8 the focus group sections were started out with a very open question. That was intended to allow the participants to get to know each other. It turned out that these discussions provided a lot of interesting themes. A mind map technique was used for the initial analysis of the first open question "*What is ERP implementations about and why are they difficult?*". This mind map was shown in the Data Analysis section 8.2, figure 8.1.

The first round of analysis just laid out the different discussions going on, followed by a second analysis where themes were extracted from the discussions and individual contributions to the discussions were related to the themes. In table 9.3 a short heading for each of them is provided.

Themes in the discussion in focus group 1	**Themes in the discussion in focus group 2**
• The scope and the nature of an ERP system has changed over time • Change management • Standardization (why standard ERP software) • Stakeholders - heterogeneous and changing over time	• Users as a complication to the ERP implementation • The consultants role • Requirements specification and managing expectations • Customizations and the scope and nature of an ERP system has changed • Stakeholders

Table 9.3: Themes that came up in the focus group discussions.

As I was reflecting on the discussion themes above; considering the perceived role of the consultants, management, user representatives, suddenly I noticed that different metaphors were used by the participant. A metaphor is a cognitive lens used by a person to make sense of situations, and is intimately interconnected with the way one thinks (Kendal & Kendal 1993).

Looking closely at the transcripts three metaphors could be identified. Technically I was going directly to the text produced and used the participants own language to identify the metaphors. The three metaphors are:

1. ERP implementation as "*a standardization war*"
2. ERP implementation as "*a game*"
3. ERP implementation as "*a change project*"

The following citation is the once that gave rise to the standardization war metaphor:

> "*It is much about standardization and integration …….. You get into trouble if you do not win the* ***standardization war****. …..… Having everybody use the same system is both the challenge and the goal*" (focus group section 18)

And others build on the metaphor e.g. explaining how the prior IT system may influence the war."*If you come from* (the customer come from my addition) *a tailor made system you* (the implementation team my addition) *are sure to be* ***defeated***." (Focus group 1 section 28)

The war metaphor can also be seen to influence the relation to the user organization, as an example the ERP consultants talk openly about taking users hostage and fighting with the user organization. "*Here we took three countries hostage you could say, in a nice way, but three countries were* ***taken hostage****"* (Focus group 1 section 56) *"In each country we had* ***a fight*** *over customizations needed because of local legislation."* (Focus group 1 section 60)

The effect of the war metaphor also seems to be visible for the internal ERP experts. In the other focus group one of the internal ERP experts are complaining about the external ERP experts' way of treading the user organization. *"I have seen consultants* ***take my user representatives as hostages****. I know it can sound sharp but some of them are not use to or tough enough to go up against consultants*." (Focus group 2 section 241).

In the external ERP experts mind the war is legitimate because standard systems are cost effective and serve a grater purpose e.g. integration and standardization "*You want one integrated system sharing master data, using the same processes and input screens…..it is all about standardization and integration. No one think that is fun*" (focus group 1 section 18), or the system works to serve management *"I was once part of a company's group management, the situation was, that we had some subsidiaries that needed to have the system* ***enforced on them****"* (Focus group 1 section 10)

Another way to legitimate the "war" is the perception of end users as being unable to see what is best for them. "*Customer XX used 11 segments in their chart of accounts, one of them the VAT code …. Basically this destroys the idea of a standard system*. (Focus group 1 section 24), and they are unable to apply a critical look upon what they do therefore they need someone from outside to open their eyes *"I also experience a wish to map the old world to the new. Often because your had no other way to handle the complexity in your old world than e.g. to build it into the chart of accounts. Now you have a large application portfolio providing functionality but…….*" (Focus group 1 section 27). Thus if the user organization is given the power it is impossible to implement a standard system. "*They try to customize the system as much as possible to the organizations way of working instead of living with the*

more limited possibilities the standard system provides. Suddenly it is no longer a standard system." (Focus group 1 section 23)

The external ERP experts perceive them self as being on *a mission* working for management (and standard systems) against the user organization. "*With vendor XX the **mission** was to define the business case and follow it through -**force it through** the organization ……. The perception was that we (us as consultants) acted as an auxiliary arm to management*."(focus group 1 section 20), and the external ERP experts are willing to stand up against the user organization "*Use standards don't develop different obscure corners to the system because the user organization claims their life depend on it*." (Focus group 2 section 13)

9.7.1 ERP implementations as a game (a stakeholder perspective)

The game metaphor could be recognized in the following citation.

> "*I think this is a problem of grate important in this **game***" (Focus group 1 section 17)

The "game" is characterized by having many different players (stakeholders) having different goals and different rules to play by, both on the vendor side and the customer side " *For the vendor it is about selling licenses. ……. at the end it is all about earning some money. For the customer it is to get a solution to some basic problems.*" (Focus group 1 section 16).

The number of participants in the ERP game is huge, and change over time, and they have different goals "M*any different goals also on the vendor side*"; *sales person, pr-sales consultant, implementation consultant, programmer, support department and after sales*.", and at the customer side *"At the customer side you have equally diverse goals. It can be one person who wants to thumbprint something. Obviously the person believes it is for the best of the company, but it is important for him that he made it happen. In the organization there may be others with completely different goals, and maybe somebody that actually will suffer from the decision.* … and as a participant in the game you cannot predict the next move of stakeholders or the outcome *"For me one of the main issues is that those involved in preparing and signing the contract is to distant from fulfilling the contract and actually taking part in the work….. []It gets more and more impossible to oversee the consequences of such a project"* (Focus group 1 section 17).

Related to this metaphor the participants emphasize that ERP systems are not just a standard system, it has several possible solutions that need to be evaluated and negotiated. "*One of the more important things about the large systems is the enormous customer base. Over the years experience is collected,…[] thus not just one way to do things but a number of variants ….[] there should always be a solution which any company can live with, or at least use and find right for them."* (Focus group 1 section 41).

Thus the focus group participants who initiated and developed the "game" metaphor emphasize the unpredictable nature of ERP implementations, and the fact that success in not a given, it depend on the stakeholders point of view. As an ERP expert you have to adjust your behavior and strategy to maneuver in this unpredictable and changing environment. However in the focus group cession the same participants who developed the game metaphor also from

time to time elaborating on the war metaphor. Although it may seem contradictory I think it make sense because they perceive themselves as one of the stakeholders (the external ERP expert), and they are given the major responsibility for getting the system up and running, as a standard system. Thus they are up against the suspicion and rejection that is inherited in change, and they are unable to accommodate all stakeholders at the same time, thus they have to find ways to handle rejection and disappointment.

Business consultants representing smaller areas cannot oversee the consequences of implementing a standard system covering the entire organization. All they do is thinking about their own area and require what will make them stand well with their own group. But it is an integrated standard system that we are implementing, they don't understand that data entered in one module is made available elsewhere. This kind of relations IT people has to help explain, plus cut down business requirements that come out of the blue. The way you have to act is questioning everything all the time. Should this really be included? Are you sure this is necessary? What do you need this for? Can't it be done in a different way? I have seen many business people requiring what is easy for them to get acceptance for in their hinterland." (Focus group 2 section 25)

9.7.2 ERP implementations as a change project

The "change project" metaphor was initiated in the second focus group in this citation:

> *"It is important to approach it as a **change project,** which is the major challenge. If you approach it as a business change project then very different mechanisms come into play from the very beginning. It creates a very different communication with the organization. The expectation you create in the organization is different compared to perceiving it simply as an implementation of an IT system. It is very, very different."* (Focus group 2 section 21)

Contributions to the "change project" metaphor came from different participants, and two slightly different perspectives could be recognized; an (organization) internal perspective and an external ERP expert perspective.

Using the internal perspective business cases (suggestions for re-engineering) are developed internal in the client organization before the ERP experts are asked to help configure the ERP software, the ERP consultants' task is to help find software solutions that accommodate the business cases in the best possible way. Using this perspective it is the organizations responsibility for the re-engineering "*Regarding responsibilities and challenges I don't think the organization can expect "a brilliant consultant" to take over the responsibility or require them to give the right advice. It is the organization's own responsibility. If you are not aware of your own responsibilities then you make a big mistake, and excuse me, but then you binge it on yourself …….[]You should rather get the right employees in key roles; have them ask the right questions, someone that know the marked and understand both IT and the structure; both advantages and disadvantages. Someone that have all-round knowledge and are able to make the connection to the business."* (Focus group 2 section 57)

Some objections are given that it may be too time consuming, and if the project have a deadline, the it might be impossible, and he reply "*It is my experience, that if you are the kind of organization that needs a tender process, then it takes time, maturing time; attitude time I might say. Top management is involved and you are turning many stones. You get into all corners and hopefully you have time to consider it thoroughly. You also have time to make a risk profile for your project, so that you start the project with your eyes open and know how to do it.* Thus it is emphasized that rushing into the configuration process might not be a god idea and that you as an organization need to prepare for meeting the external consultants. "*Instead of starting the project to quick, because you want to finish soon and then shooting yourself in one foot. Becoming a project manager in that situation is not good, because you get into discussions that actually should have been taken upfront while making a thorough analysis including a business case. That is very important to have it done that way."* (focus group 2 section 59)

The external perspective emphasized that the ERP consultants is working for management and is expected to develop suggestions for re-engineering the organization's business processes and justify them with business cases. This is done prior to the configuration and customization, or during configuration and customization. This perspective has a very strong focus on the ERP software. "*When I was working for xx (consultant company implementing ERP) then you had the idea that what was driving the project was business cases. Thus your mission* (as an ERP consultant) *was to develop the business case and follow it through the organization. On as large a scale as possible……[]you* (as an ERP consultant my addition) *worked for management.*" (Focus group 1 section 20)

Common for the internal and external perspective on the "change project" is that change can be planned and executed. *If it* (the ERP project my addition) *is suppose to change and develop the business then it is much process oriented. You have to be open to what kind of project…. It is very common the organizations have fear of contact, they won't face the consequences. They would like the result, but it comes with a cost. It is here the business case comes in; what do we gain and what will it cost? …. You have to know it and plan accordingly! "* (Focus group 2 section 24)

9.7.3 The three metaphors and the perceived role of different stakeholders

The three different metaphors have implications for the perceived goal for the implementation and the perception of different stakeholders' role in the implementation. In table 9.4 the complete result of the analysis is presented (in appendix B section four more details related to the empirical material can be found).

Metaphor	Perception of the consultant's role	Perception of the system's role	Goal for the implementation	Perception of the role of management	Perception of the role of design team	Perception of the end users
A standardization war	• Defending the (IT-) system • Working for standards (management)	• Best Practice (standard) • Common infrastructure	• Integration on all levels • Visibility (data and business processes) • Common, operation and maintenance of the IT-system	• Deciding on the ERP system • Defending the standard	• Configuring an organizational wide standard IT-system	• Adapting to the standard • Passive – receiving training in the new IT-system
A game	• Serving a specific stakeholder in the organization (typically the IT-department) • Recognize stakeholders on both side change over time ➔ the game changes over time • Guarding the standard system	• Supportive infrastructure (hosted by IT-department) • Cost effective IT-solution • Work for all organizations	• Stakeholder satisfaction • Minimize cost (and effort) developing and maintaining IT	• An important stakeholder • Sponsor • Court of appeal regarding design decisions and conflicts between stakeholders • Support standard IT-system for cost reasons	• Negotiating design suggestions • Focus on the IT-artifact	• A stakeholder • Able to directly or indirectly influence the design team and the final design • Able to influence other stakeholders • Receive training in the new IT-system
A change project	• Working for the design team as "experts" (process designers) • Develop design suggestions based on the business case/requirements specification deploying the standard system as much as possible	• Designing the IT-artifact is secondary to the change process • As much reuse (standard) as possible	• Changing the way the organization work • IT-support to fulfill the business case • Integrated and optimized processes	• Develop or approve business cases (ensure strategic fit) • Communicate why change is important • Ensure resources for the project	• Strategic focus (IT-artifact secondary) • Design experts able to optimize the business and work processes (could be either internal or external experts) • The design teams work can be guided by the business cases	• Informants • Need to be included for psychological reasons and to some extend for knowledge diffusion • Receive training in new (socio-technical) processes

Table 9.4: The complete analysis of the three metaphors.

9.8 Summing up - answering the detailed research questions

The first research question for the focus group study was:

2a) How do ERP professionals perceive ERP implementations; what are they about?

As an answer to this question three different metaphors were identified (described in details in section 9.7):

- ERP implementations as “a standardization war”
- ERP implementations as “a game”
- ERP implementations as “a change project”

The second research question for the focus group study was:

2b) How do ERP professionals perceive the need for knowledge integration in order to design the ERP system?

The analyses show that the ERP experts in general perceive it to be very important for the ERP consultants to obtain knowledge about the organization. The knowledge is seen to be important in order to map the organizational processes and the ERP software processes to each other. But different arguments are used depending on if a consultant or customer perspective is applied.

Seeing it with a consultant’s perspective the knowledge is important in order to be able to interpret the requirements and understand where the organization is coming from. In general it is agreed that understanding patterns of the processes are sufficient, and if possible you should focus on to-be instead of as-is. If it becomes a matter of trying to understand how an individual user perform a specific task then you have come to fare, you need to stay on an abstract level. It is assumed that requirements are derived from the existing practice, and that it is necessary to understand the user representatives’ “social world” to some extend in order to fully understand the requirements the user organization put forward. The knowledge obtained is however not intended for become sympathetic with what is going on in the organization; on the contrary. The knowledge about the organization is considered important in order for the ERP consultants to argue with the client organization why the standard software is better than the existing practice or at least useful for the organization. Thus it is necessary in order to be able to argue why the standard solution can be (or should be) used in the organization.

The focus group participants coming from user organizations emphasizes that they think it is important for the ERP consultants to get to know the organizational practice and the organizational context in order to fully understand the unique characteristics of the organization. Thus, it is important in order to understand why the standard solution cannot be used in the organization, and in order to map organizational processes to the ERP system not the other way around.

Although the two perspectives seem to agree that you need to map the ERP systems processes and the organizational processes to each other, they disagree on whether you take a starting point in the organization's processes or the ERP system's processes. This might seem unimportant, but it makes a difference if you are trying to understand what to change. If you take a starting point in the ERP system's processes and discus why and how they are problematic for the organization, then "changes/customizations" necessary to the ERP system's processes will be specified as a natural consequence of the discussions. Derived consequences in regard to the organizational work processes will however become hazy.

The metaphors are found to influence the perceived level of detailed knowledge the ERP consultants need about the organizational practice. Thus the participants subscribing to the "change metaphor" are assuming that the ERP experts are responsible for process re-engineering, and in order to perform re-engineering more detailed knowledge about existing processes are necessary – not to make the system work that way, but to ensure that the re-engineered processes are providing the necessary outputs. Thus the importance of organizational knowledge is emphasized.

Trying to see the knowledge integration problem from the other side; asking the focus group participants how much the organizational representatives should understand about the ERP system the analysis also discover some disagreements. Again the disagreements are related to the internal-external perspective, and depend on the metaphors the participants subscribe to.

The focus group participants using a consultant perspective (especially those subscribing to the *war metaphor*) have a very firm understanding of what the organizational participants should contribute to an ERP project. Thus organizational representatives need to contribute first and foremost requirements and it is assumed to be unproblematic to provide these requirements, but they also need to provide test cases, perform the tests and perform end user training. To be able to perform tests and end user training they need to understand the solution not the technological options (the ERP software's capabilities). The organizational representatives are to a large extent perceived as be "super users" that just need to know about the system (the final solution) a little before the rest of the organization in order to be able to spread the knowledge in the organization.

The participants using the *game metaphor* have a little different perception; they understand the organizational representatives to be stakeholders, which need to be included in exploring different technical solutions while negotiating them in the organization. Thus the idea of an "absolute" and "objective" requirements specification is not as dominant as in the two other metaphors. They acknowledge that requirements may depend on the stakeholders and that the solution need to be negotiated in the organization, thus the organizational representatives and the ERP experts need to cooperate deciding on the solution and therefore the user representatives need knowledge about the ERP systems capabilities.

The external ERP experts subscribing to the *change project* metaphor consider business cases and the belonging re-engineered processes the "requirements", and they understand the ERP experts to be those performing the design. Thus knowledge integration in the sense that organizational representatives need to understand the ERP software's capabilities in order to participate in deciding on different obtions is considered of limited importance. Discussions with the organization are perceived to stay on a more abstract level; end-to-end processes.

However, testing and end user training is still perceived to be a job for organizational representatives and in order to perform these tasks they need knowledge about the solution not general knowledge about the ERP systems capabilities.

The third research question for the focus group study was:

2c) Which design artifacts do ERP professionals use and why?

In figure 9.1 the initial evaluation by the focus group participants are shown. The detailed analysis (section 9.2) of the discussion that followed after grading the techniques provides many details and show that the differences in evaluation often depend on if the participant has a background as external ERP consultant or is internal to an organization. But also differences related to the metaphor the focus group participant subscribe to. Especially three techniques are in general perceived to be useful: visits in the user organization, examples of documents used in the organization prior to the ERP implementation and discussions with organizational representatives.

	1	**2**	**3**	**4**	**5**
	Not useful		Could be useful	Very useful	
Visits	XX			XXXXXX	X
Documents		X	XX	XXX	XXX
Requirements specification	X	X	XXX	X	XXX
AS-IS data model	XXX	XXX	X	XX	
AS-IS process descriptions	X	XX	XX	XX	XX
Use cases		XX	XX	XXXX	X
Discussions			X	X	XXXXXXX

Figure 9.1: Initial evaluation of techniques by the focus group participants

Additional techniques added by the focus group participants (details in section 9.3):

- Workshop based on demo data

- Interview external stakeholders
- Management interviews
- Legislation and e.g. salary agreements
- Demo of exiting IT system(s)
- Externally provided TO-BE processes
- Rich Picture

Although the focus group participants find these additional techniques very useful they are not always used. Interview with external stakeholders, interview with management, legislation and externally provided to-be processes are all added by participants subscribing to the *change metaphor*.

In figure 9.2 the initial evaluation by the focus group participants are shown. The detailed analysis of the following discussion provide many details (section 9.5) and show that the differences in evaluation often depend on if the participant has a background as external ERP consultant or is internal to an organization. But also differences related to the metaphor the focus group participant subscribe to. Standard education and discussions with the consultants are perceived as the most useful techniques.

	1	2	3	4	5
	Not useful		Could be useful	Very useful	
Standard education			XX	XXXXXX	X
Explore the system them selves	X	XXXXXX	XX		
Walk through process models		XX	XXX	XXX	X
Sitting side by side with the consultant exploring screens	X		XXX	XXXX	X
Discussions with the consultant			XXX	XX	XXXX

Figure 9.2: Initial evaluation of techniques by the focus group participants

Additional techniques added by the focus group participants (details in section 9.6):

- Internal demo before and during implementations
- Workshops or prototypes on live data
- Early training and training in new processes
- Collaboration lab
- Best practice accelerators

Common for all the additional techniques suggested is that they are very costly and time consuming therefore the focus group participants find that they are very seldom used, but if used they consider them extremely effective.

The primary shared design artifacts used are: a traditional requirements specification, internal and external documents used in the organization, the ERP software used as an emerging prototype (sometimes accompanied by abstract descriptions of to-be processes), and especially if a *change metaphor* is applied business cases (including specifications of the re-engineered processes).

Chapter 10
Concluding on the second learning cycle

In this chapter I summarize the result of the second learning cycle, and provide a background for understanding the research performed in the next learning cycle.

10.1 What did the second learning cycle contribute (what do I know now)?

- I know that ERP professionals perceive ERP implementations in different ways identified as three metaphors: “a standardization war”, “a game”, and “a change project”.

- I know that some of the participants contribute to more than one metaphor: slide in and out of a metaphor e.g. referring to different projects or working for different consultancies.

- I know that the war metaphor is very dominant among external ERP consultants participating in the focus groups, and even if subscribing to another metaphor a participant very easily agree to arguments belonging to the ware metaphor; they even use such arguments themselves from time to time.

- I know that these different perceptions (metaphors) result in very different expectations to the stakeholders around ERP implementations.

- I know that design of organizational processes is central to the differences. Who or what is designing the organizational processes how?

- I know that the participants have somewhat different opinions about which techniques and tools to use in order to obtain knowledge for design decisions. Some of the differences can be related to the different metaphors. Thus in some areas I have establishes an understanding of the relation between different perceptions of ERP implementations and the approach used for the cooperation with the user organization.

- I know that the more used tools and techniques (and the way they are used) result in a focus on the ERP experts’ needs, thus the consultants need to obtaining knowledge in order to configure the system has high priority.

- I know that the focus groups participants have positive experience with different techniques and tools (internal demos before and during the implementation, workshops or prototypes on live data, collaboration lab, and best practice accelerators) that allow the organizational participants to develop knowledge about the ERP software’s capabilities and the suggested solution. However it is the focus group participants’ experience that such techniques and tools very seldom is used; costs seem to be the argument for not using them.

10.2 A second go at the answer to the thesis' overall research question

The overall research question for this thesis is: how do organizations engage ERP implementations and why does it often result in misfits? Considering the findings in this second learning cycle a preliminary answer could be.

There may be many different ways to engage ERP implementations as the three metaphors identified indicate.

Misfits may be a natural consequence of the approach used; systematically rejecting to consider the existing organizational practice.

The external consultants dominate perception of ERP implementations as war may also provide a somewhat tense relation between the ERP system and the (user) organization.

The approach used before go-live to obtain knowledge how to design the system seems to be very likely to result in logical/factual defects in the design of the new ERP software (not considering indispensable requirements given by the organizational context), and provoke a reaction toward the system as soon as the ERP consultants leave the stage.

10.3 Where is the research going?

The focus group study drew my attention to the different perspectives on work process design. For a long time I had focused on process design as stakeholder negotiating (inspired of what played out in the case study). The focus group participants however made me realize that I completely had neglected that business process re-engineering and "best practice" business processes was a very important element in the rational arguments for adopting ERP package software.

But how does work processes come about and how can the relation to the IT artifact be considered?

Work processes could be understood to develop over years as a result of stakeholder negotiation; as a result of an articulation processes (Strauss 1988). Stakeholder negotiation seems to play a role in ERP implementations, but it is not really this way process design is anticipated in relation to ERP implementations. However stakeholders' power, stakeholder relations, and stakeholder buy into the solution somehow seems to be important to consider.

Slowly I realized that traditional arguments such as not changing the standard system because it is expensive and make upgrades more difficult, or the argument that ERP package software is an easy way to realize best practice business processes, was in the way of developing an understanding of what ERP implementations actually is about. As I see it, it is about coordination; coordination in heterogeneous and distributed environments. Segregating and standardizing processes makes coordination easier; it rationalizes coordination, but it requires explicit design work (Gerson 2008).

The case study had me started wondering about the coordination aspect (coordination simply sharing data, complex coordination supported by work-flow), and the importance of having users understand how to perform their part of the process in order to provide coordination and high quality data. The focus group study made it obvious that different people had different perspectives of whom/what were expected to be responsible for process design including design of coordination mechanisms. I came to think of an old paper arguing that a new group of designers; business system designers need to be included in information systems development (Markus and Keil 1994). But who are these process designers and what are they doing, and how does it relate to business processes as they are performed in organizations?

Furthermore, how are the processes related to the ERP software? All these questions made my think and look into different theories that could help me make sense of all this.

Thus a third learning cycle was engaged.

Part IV – The third learning cycle

Part IV - *The third learning cycle about re-conceptualizing ERP implementations* contains four chapters.

Again I start with a short discussion and presentation of the research method; that is chapter 11.

Then in chapter 12 I present articulation theory as a theoretical framework and argue how ERP implementations can be seen as an articulation process.

In chapter 13 I use the misfit categories developed in chapter 5 to show why the process perspective normally applied in articulation theory needs to be accompanied by a taxonomic perspective in order arrive at the necessary level of data "standardization" needed for coordination as well as decision support on all levels in the organization.

In chapter 14 I finally present a theory for re-conceptualizing an ERP implementation as an *ERP articulation process*. The fourth learning cycle is completed with the ERP articulation theory.

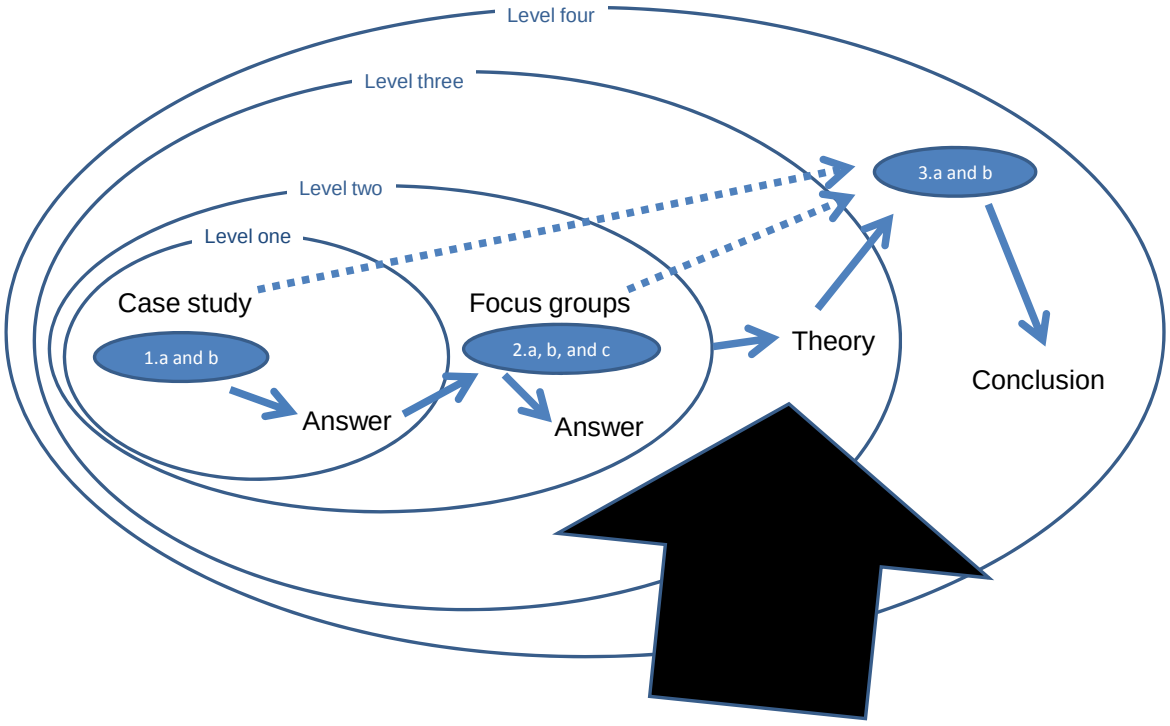


Figure IV.1: The overall structure of the Ph.D. thesis with a thick arrow indicating where part IV belongs

Chapter 11
Research Method for the 3rd learning cycle

In this part of the thesis I develop a theory for understanding ERP implementation processes as they take place when organizations adopt ERP package software.

The approach I used to develop the theory is as follows. First I read literature about articulation theory (a short introduction to the essential theoretical constructs is provided in section 12.1). Then I considered how to elaborate the theoretical constructs to the area of ERP systems (in section 12.2 a preliminary understanding of ERP implementations seen in the context of articulation theory is provided).

Then in chapter 13 I use the misfit categories from chapter 5 to understand whether the ERP articulation process can explain all the misfit categories. It turns out that the fourth misfit category (taxonomic misfits) isn't covered by the preliminary ERP articulation theory thus a new construct has to be added.

In chapter 14 then, my final theory for an ERP articulation process is presented.

In figure 11.1 below the steps used in developing the theoretical framework is illustrated.

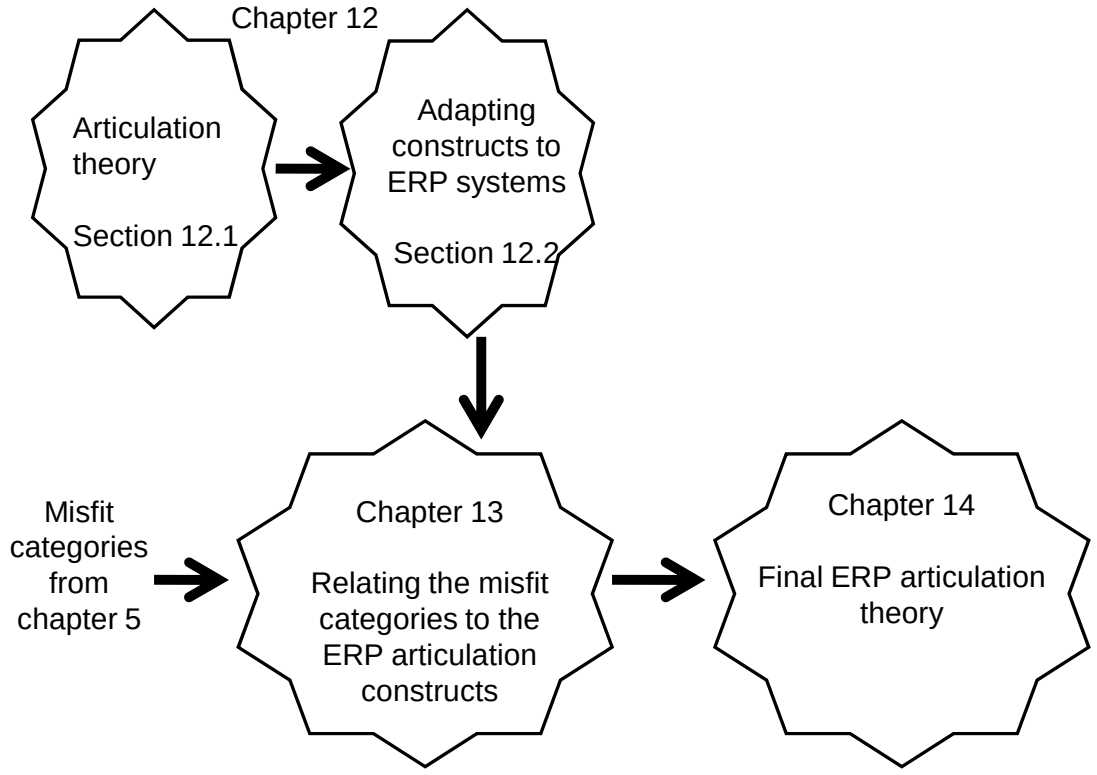


Figure 11.1: The structure & relationship between chapters in the 3rd learning cycle

Why choosing articulation work as the theoretical basis for my theory building? The result of the focus group study made me realize that different perceptions of how work process design is approached played an important role in the metaphors applied by the ERP experts. The case study also show that it is seems reasonable to perceive ERP implementations as a design process with dual design objects; the IT artifact and the organizational processes. However I haven't seen any ERP research trying to explain this interwoven nature of the design process.

This really got me started wondering how organizational processes come about and if they can be designed. Being able to design people's behavior seems unreasonable to me as I believe all humans have some degree of freedom to act, but anyhow I started considering different theoretical lenses.

Some years ago I participated in a PhD course given be Elihu Gerson and suddenly I remembered that I had been very fascinated by a paper he gave us to read (at that time unpublished but later it became a book chapter (Gerson 2008)). What had fascinated me was the idea of Information Technology (the internet) understood as an abstract protocol used to coordinate the actions of different people; actually without having the actors explicit agreeing how to do it. Coordination was possible simply because one part published information on the internet that allowed the other part to adjust his actions accordingly. Thus I decided to explore if Gerson's work could help me understand what was going on.

Returning to the paper I realized that it was based on articulation theory, originally introduced by Strauss (Strauss 1988), and it turned out that articulation theory actually provided a very useful theoretical foundation for understanding how business processes come about. I also realized that articulation theory over the years had been used and elaborated within information systems research e.g. (Gerson and Star 1986; Schmidt and Simone 1996; Baker and Millerand 2007).

Gerson's idea introducing a protocol as a construct in the articulation theory provided the link I needed between the articulation processes and the IT artifact (the design of the IT artifact).

In chapter 12 I explain the constructs of articulation theory and how they can be adapted to ERP systems implementation.

In chapter 13 I return to the misfit categories developed in chapter 5, and use them to illustrate how different misfit categories relate to the articulation process. The analysis provides more detailed understanding of ERP implementations as articulation processes, and it makes it visible that a new perspective on the articulation needs to be included, a taxonomic, perspective.

In chapter 14 the complete theory of ERP implementations as an articulation process (revised according to the findings in chapter 13) is presented.

Chapter 12
Articulation theory, coordination & reconciliation

In this chapter I introduce articulation theory as a lens for understanding the ERP implementation process that takes place when organizations adopt ERP package software.

This chapter is structured as follows: In section 12.1 I introduce the theory about articulation of organizational processes, in section 12.2 I argue how existing constructs can be elaborated to cover the context of ERP implementations.

12.1 Articulation theory

Taking a more general perspective on how organizational processes come to be the notion of articulation processes can be used. When people are working together in organizations or across organizations they need to find a way to organize the work allowing different actors to perform their part of the cooperative work process. Thus some kind of organizational work patterns have to be negotiated in order to arrive at the processes used to carrying out work. Using the terminology of Strauss (1988) arriving at organizational work arrangements can be understood as an *articulation process* which embraces the general organizational process and the phenomena that constitute it.

Formal protocols and procedures for carrying out organizational processes may exist, but since no central authority can anticipate everything that might happen in local settings, then local actors will always have some discretion in relation to how to enact a protocol or a procedure in order to get the work done in a way that accommodate different stakeholders' needs and points of views. The need to modify a formal procedure could also result from changes in the context, thus formal procedures may easily become slightly out of date as changes in open systems happens all the time; changes in actors, technologies, alliances, products, etc. Thus organizational work arrangements needs constant customization to accommodate such changes or putting it another way; constant negotiation of changing patterns of task organization is needed.

Strauss (1988) argues that analytically it is useful to be able to distinguish between two aspects of articulation: *the articulation **process*** and the *articulation **work.*** Splitting articulation into two different constructs allows an important distinction; the *articulation process* focus on how work is actually performed by actors while the *articulation work* has a more descriptive or prescriptive nature focusing on "*the specifics of putting together tasks, task sequences, task clusters*" – even aligning larger units such as lines of work and sub-projects – in the service of a workflow ((Strauss 1988) p. XX).

Gerson (2008) makes another distinction: *Local articulation* and *metawork.* This distinction is based on Strauss' original difference between *articulation process* and *articulation work* explained above But the original constructs are given new labels to emphasize the difference between articulation in a particular situation (local articulation), and articulation as specifying what goes into a work process (metawork). This distinction is important for Gerson in order to

address different aspects of articulation work in situations where work is distributed in time and space.

Gerson's work also introduces a new and important construct that is important in relation to ERP implementations namely the concept of *protocols*. A process protocol can be understood as a procedure or a formal description of how to perform a process. Using protocols to support work processes emphasizes metawork; specifying what goes into a work process. If process protocols are used in distributed and/or heterogeneous environments then metawork may be necessary both at the global and local level. Thus adapting the general (global) to the local at multiple levels becomes a new aspect of articulation.

12.2 Coordination mechanisms

When considering how to coordinate people and tasks in cooperative work processes the notion of coordination mechanisms plays a significant part. The basic problem is how to bring resources and results together at appropriate times and places in the work process. Schmidt and Simone (1996) have defined coordination mechanisms as distinct artifacts e.g. bills of lading or restaurant checks.

*"A **coordination mechanism** is a specific organizational construct, consisting of a **coordinative protocol** imprinted upon a **distinct artifact,** which in the context of a certain cooperative work arrangement **stipulates** and **mediates** the articulation of cooperative work so as to **reduce the complexity of articulation work** of the arrangement (emphasis in original)." (Schmidt & Simone 1996, p. 180)*

An alternative way to think of coordination mechanisms is to consider them "*tasks dedicated to orchestrating the work of other tasks"* (Gerson 2008, p. 4). This notion of a coordination mechanism is recognizing that both the artifact and the work of using it is important. Thus a bill of lading will only serve its purpose as a coordination mechanism if everyone is using it the intended way. Following Gerson's line of thoughts the work preparing and using a coordination mechanisms correctly is a part of the mechanism and should be considered a specialized kind of task. Gerson's notion of coordination mechanisms is the more inclusive of the two definitions so I decided to use that in this thesis.

Coordination mechanisms can also be understood as a mean to reduce the complexity of metawork and as an effective mechanism to *rationalize* the interaction of the coordinated tasks.

Coordination mechanisms *"are forms and procedures used to reduce the complexity of articulation work; that is rationalize it by making parts work together smoothly" (Gerson 2008)*

Thus if rationalizing is understood as doing more with the same resources or doing the same with fewer recourses then standardizing and segregation of work processes can be expected to provide rationalization. Establishing procedures or protocols for organizational processes may complicate or constrain situated articulation but at the same time provide the ability to coordinate across larger organizations and distributed work arrangements (Gerson 2008).

Standardization and segregation is expected to provide rationalization by:

- Replacing specialized and situated knowledge with general procedures (e.g. using local process protocols to make situated articulation less dependent on specific individuals with specialized knowledge)
- Make things work in comparable ways in many situations (e.g. use the same local process protocol in multiple instances of situated articulation)
- Reducing or removing the need to know how others have arranged their work if everyone works according to the protocol (e.g. make situated articulation less dependent on situated work arrangements elsewhere)

12.3 Reconciliation

Arriving at agreed (accepted) arrangements of coordination (design of coordination mechanisms) often require serious negotiations, or maybe actually battles, since a coordination mechanism often have derived consequences for the work performed on both sides of the mechanism. Multiple contending stakeholders may have different viewpoints, interests, and concerns in relation to the actual design of the coordination mechanisms. Thus reconciliation becomes an important aspect of the articulation process, and supporting reconciliation during articulation work means providing tools and procedures for working with complex negotiations. Articulation aimed at reconciling differences cannot be formalized because it involves justifying, designing, choosing, and enforcing the articulation procedure in itself (Gerson 2008).

Reconciliation is expected to be more complicated if no common authority exists. Gerson gives three examples of organizational arrangements that may come into play when no common authority exists:

1. Cross-cutting ties – alternative network crossing formal boundaries of departments or sub-divisions
2. Participant review – more useful in situations where the relevant evaluation criteria are clearly understood and reproducible, and when raters have a stake in the quality of the outcome
3. Patronage – personal relations between a few relatively powerful patrons and their relatively weaker clients

12.4 Adapting the articulation theory constructs to ERP systems

In relation to ERP implementations it is important to be able to make a distinction between articulation as something actually happening when people do their work, and articulation as an activity oriented toward specifying and negotiating procedures. Both Strauss' original distinction between the *articulation process* (actually doing) and the *articulation work* (putting tasks together in sequences), and Gerson's distinction between *local articulation* and *articulation work* is important. Strauss' distinction draws the attention to the difference between how work processes are described/prescribed in contrast to how they are actually performed (enacted). Gerson's distinction on the other hand draws the attention to the notion

of "*local*". In a highly distributed work environments what is local? A work process (work flow) can be distributed across multiple geographically spread units and still be considered "local" in some sense. Thus local may involve differences in actors, differences in their culture, differences in their environment, and differences in organizational belonging. In the context of ERP systems "local" easily becomes very complex and the articulation process therefore rather complicated due to, for example, differences in goals and knowledge bases, and due to the lack of common authority.

To be able to address both Strauss' original distinction between the articulation process and the articulation work, and the complication of "local" in Gerson's distinction I need constructs that allow for a more elaborate definition of local and at the same time allow the distinction between the articulation taking place at a specific time and place, and metawork.

Hence for the unity of all aspects of articulation related to ERP implementations I will use the construct *articulation process*, and to address specific *articulation layers* of the articulation process I will use the following constructs:

- *Situated articulation* will be used to address articulation taking place in a specific instance of time and place. Situated articulation can also be understood as articulation "in use". Thus it involves adapting a local protocol to situated use; humans enacting local process protocols.
- *Local metawork* will be used to address the development of a local protocol or adapting a generic local protocol to a context specific local protocol. Local protocols may be intended to embrace multiple instances of *situated articulation* distributed in time and place, and/or organizational belonging.
- *Federal metawork* will be used to address the development of a complex of protocols for a specific federal context (e.g. an organization or a coalition of organizations) or adapting a global protocol to a specific federal context. Federal metawork is focusing on the design of mechanisms that tie processes together in an integrated (federal) complex of local protocols.

I use the term *articulation layer* to emphasize that no order or importance can be ascribed to the layers. They all have to be performed to complete an articulation process and they are preformed simultaneously or iteratively. The distinction between situated articulation and metawork (both local and federal metawork) however allows me to address the design aspect of work processes without anticipating that it is possible to design human behavior.

In the context of ERP implementations (articulation) preparing coordination mechanisms is a natural part of metawork (both federal and local metawork). Hence understanding how to standardize and segregate, and still be able to coordinate tasks is the very essence of the work performed during metawork. But as emphasized by Gerson (2008) a coordination mechanism will only serve its purpose if people are able to use it the intended way during situated articulation. The more heterogeneous and distributed environment local process protocols have to cover the more challenging it become to think out ways to ensure that conceptual schemes are aligned both when designing coordination mechanisms and when using them.

An example of this issue could be:

- Someone in an ERP implementation specifies a purchase procedure (sequence of steps and rules allowing a purchase order to move between steps), and in addition specifies how this procedure should be reflected in the design of the ERP system (some steps in a work-flow, data definitions and a user interfaces for situated articulation). We are now anticipating that employees in different departments in the organization (maybe even placed in different countries) have to use the ERP software's user interface to order any kind of goods simply by filing data into the purchase order screens.

- The software "knows" the procedure (the protocol) specified for purchasing in this organization and if every actor fills in screens in the anticipated way then they don't need additional communication. Thus the ERP software is perceived to have two important functions in relation to coordinating work in the organization (1) It "knows" about the specified organizational process procedure (the coordinative protocol), and (2) the user interface will guide and support the user (situated articulation) doing his part of the process in the anticipated way.

- Designing a local protocol (the purchase protocol) and user interfaces that will support many different users perform situated articulation in a way that allows the ERP system to have a coordinating effect isn't simple. The designer have to anticipate (and include) different situations the local protocol could/should be used in, e.g. different types of goods and services, different physical and cultural environments, differences in skills, sporadic users and professional users.

As explained above work arrangements (work processes) in distributed and heterogeneous environments as those ERP package software are supposed to support can be understood as a complex of work processes tied together by coordination mechanisms, and ERP package software can be seen as a protocol for such a work arrangement; a detailed plan of procedures.

The word *"protocol"* may have a specific meaning in specific communities e.g. as in is used within data communication, but here the word protocol is used in a very general sense. Thus a protocol could just be a guide for how to do something as indicated in this citation *"A protocol is a rule which guides how an activity should be performed. ...[] Protocols specify the proper and generally accepted behavior."* (Wikipedia 201008 Dr. P.M. Forni on behalf of the International Association of Protocol Consultants)

The ERP package software fit the understanding of a protocol in the sense that "best practice" work processes implemented in the software can be seen as reflecting well established and time-honored rules that make it easier to work together, or it could at least be seen as established patterns of operational work arrangements.

Implementing ERP package software in an organization can be understood as an articulation process where the process protocol build into the ERP package software is adapted to the specific (federal) context.

The ERP process protocol provided by the ERP package software is expected to reduce the effort of federal and local metawork. Of cause that is dependent on the ERP process protocol's ability to "fit" or replace prior federal work arrangements. Thus when adapting the ERP protocol to the organizational context two attributes of the ERP protocol is especially

important; (1) the completeness of the ERP protocol and (2) the degree to which the ERP process protocol allows customization. These two attributes however can make each other obsolete; if the ERP protocol is complete no customizations are necessary, and if it is easy to customize the ERP protocol then it is not critical whether it is complete.

As my empirical findings show the ERP process protocol provided by the ERP package software adopted by the case organization wasn't complete. Thus although adapting an ERP process protocol may reduce the effort of federal and local metawork it is still necessary to perform an articulation process including all the articulation layers in order to arrive at a useful and complete federal process protocol.

When adapting a protocol to a specific context especially two aspects of the articulation process are challenging (Gerson 2008):

1. Finding means of recognizing and describing when conceptual schemes are out of line with one another
2. Deciding on how to customize the protocol to fit the specific context making

I will get back to these two issues in chapter 14 when presenting the final ERP articulation theory. In the next chapter (chapter 13) I will apply the constructs of articulation theory explained in this chapter to my empirical field material; in the concrete I investigate how the misfit categories correspond to the different articulation layers in the articulation process.

Chapter 13
Re-interpreting the misfits

In this chapter I am reinterpreting the empirical findings from chapter 5; the misfit categories. The aim is to illustrate how the different misfit categories correspond to the articulation process.

The reminder of this chapter is structured as follows: In section 13.1 I explain how the different misfit categories relate to different articulation layers in the articulation process. In section 13.2 – 13.5 the implications of each of the four misfit categories is discussed. Finally in section 13.6 I summarize how the analysis of the misfits experienced in the user organization after go-live contribute to my understanding of the ERP implementation as an articulation process.

13.1 Relating the misfit categories to the articulation process

As explained in chapter 12 an ERP articulation process can be understood to have three articulation layers:

- Situated articulation
- Local metawork
- Federal metawork

Situated articulation could be understood to be the process where end users actually perform their work; the situation where the end users interact with the ERP software in a particular instance of time and space while performing their work. Thus in relation to ERP implementations it constitutes the "use mode" of the ERP software. However situated articulation may include an element of situated design, as situated users always have some discretion how to enact a protocol or an IT artifact. But in this chapter I will focus on the understanding of situated articulation as the use mode of the software.

Bothe *local metawork* and *federal metawork* could be understood to belong to "design mode". In these two articulation layers process procedures (process protocols) are specified and negotiated. Again design mode and use mode may be difficult to separate because changes in the context may require stakeholders to constantly re-negotiate and re-design their process protocol while they work. Here I will however consider local metawork and federal metawork as belonging to design mode, as I need to be able to address and separate design of the organizational process protocol being inscribed in the ERP software from the use of the ERP software.

The misfits experienced in the organization after go live (the detailed data material can be found in appendix A and the analysis of the misfits in chapter 5) is all derived from testimonies given by the user organization explaining difficulties doing their work and using the ERP system to support it. Thus you might claim that they all belong to situated articulation, however if you focus on who is experiencing the misfit in the organization and

the character of the misfits, then it become possible to relate them to different aspects of the design process (table 13.1 show the four misfit categories)

Categories' characteristics:	Category 1	Category 2	Category 3	Category 4
Misfit experienced by:	Individual user	Functional groups	Across functional groups	Organizational wide
Focus :	Interplay between human and technology	Difficulties performing work as a functional group (professionals)	Coordination mechanisms	Common taxonomy

Table 13.1: The four misfit categories.

Misfits experienced by the individual users (category 1) is not directly related to the design of the process protocol, they are highlighting that the human-technology interface is causing difficulties (the support for how to perform situated articulation in a way that allow the process protocol to have a coordinative effect). Thus the conceptual schemes used in design mode and use mode are out of line. In section 13.2 the difficulties are explained in more details.

Misfits experienced by functional groups are primarily related to the design of the local protocols, thus indicating that the local metawork is somehow incomplete. In section 13.3 the difficulties are explained in more details.

Misfits experienced by more functional groups are primarily related to the coordination mechanisms used to provide integration (coordination) between local protocols. Thus here focus is on coordination mechanisms that are shared by two or more functional groups. This design aspect is part of the federal metawork. In section 13.4 difficulties experienced in the user organization are explained in more details.

Misfits belonging to category 4 are however not directly related to any specific layers in the articulation process but could somehow be applied to all three layers. However, although the three layers of the articulation process require some degree of shared data definitions and shared semantic understanding, misfits in category 4 imply a need for a shared semantic understanding of data that goes far beyond what is necessary for operational process coordination. Thus misfit category 4 draw the attention to a need for *standardizing data* in order to accommodate work processes around project control, financial control, planning and more general decision support. As will be discussed in section 13.5 below, category 4 implies that when considering ERP articulation work then articulation of operational work arrangements isn't sufficient. A different kind of articulation work has to be considered in

order to accommodate the need for management information on all levels in the organization. Hence a new kind of articulation work has to be included aimed at reconciling different stakeholders' needs and viewpoints to provide a common data model and ensure a shared semantic understanding of data for management processes.

13.2 Misfit category 1: Implications for the articulation process and HCI design

In misfit category 1 three sub-categories were found, and they have slightly different implications in relation to the situated articulation, but they are all related to the interplay between the system and the situated user.

The first sub-category indicate an experienced (realized) need to customize the ERP system, not because of difficulties understanding how to use the system or because of conflicting points of view among the situated stakeholders, but simply because it is obvious to the situated users that by customizing (adapting) the user-system dialog a rationalization potential can be realized. Among the case organization's members it is a general perception that the presentation of data on the screens more or less depend on the underlying data model; a screen for each entity and/or relation. Thus customizing the screens e.g. providing data of more immediate importance for the situated use makes it possible to leave out several costly (time consuming) steps in a dialog. Other examples are related to carry over "looked-up values" to the screen where it is needed, such customizations also reduces the amount of steps (screens) the user has to perform, and in many cases also removes the need to write down the "looked-up value". These kinds of customizations doesn't seem to have any implications for the local or federal process protocol as it is only a matter of rearranging the presentation of (globally well defined) data and/or rationalizing the steps in the process performed by the situated user.

The second sub-category provides examples of situations where the situated user give up (using the system) or introduces poor data quality simply because they don't understand what data to enter into the system or how to interpret the data they are provided by the system. Thus this sub-category indicate that conceptual schemes are (has been) out of line and/or that no appropriate templates are provided to support the interpretation. Problems of interpretation will be relatively less if conceptual schemes are known to all concerned and conflicting point of views among the stakeholders are reconciled.

The third sub-category provides misfits that are not necessary isolated to the ERP software and the process protocol provided by the software. The misfits are related to obtaining cohesive force between the ERP system, other IT systems and manual procedures. Thus they imply that when deciding on local process protocols (local meta-work) it is necessary to consider other elements than the process protocol provided by the ERP software; thus alternative solutions provided by the ERP system have to be evaluated and negotiated considering the interplay with other elements in the context of situated use.

13.3 Misfit category 2: Implications for local meta-work

Sub-group 2.1 in misfit category 2 provides examples of misfits related to tasks, and sequences of tasks performed by specific groups of professionals; different communities within the case organization. The misfits they experienced are:

- Tasks that are needed in a specific situated articulation are not recognized by the local protocol
- Mandatory but dispensable (in the situated articulation) tasks included in the local protocol
- Redundant tasks has to be performed in a specific situated articulation

The experienced misfits are partly caused by indispensable requirements provided by the specific organizational context, partly by the need to adapt the local process protocol and the user interface to different use situations. Thus as expected I find difficulties related to:

- Incomplete customization (maybe not realized need) in order to adapt the ERP process protocol to indispensable requirements provided by the specific organizational context
- Incomplete customization and reconciliation due to complications of the notion of local protocols/situated use: In the case organization the local protocols; local to the complex of the process protocols, has to be used by different communities and/or users with differences in their backgrounds (IT maturity, experience with integrated enterprise systems, education, needs, power position and much more). Furthermore the same local protocol/user interface has to be used for a phenomenon which experiences considerable differences in the way it unfolds throughout the organization; e.g. differences in (amount of) data needed to depict the phenomenon and perform tasks related to the phenomenon, or significant differences in the volume of instances of the phenomenon performed in the same time frame.

Hence the misfits in sub-category 2.1 have different implications for local metawork.

- The first issue is related to recognizing the need to customize the general process protocol and the data model provided by the ERP package software in order to accommodate indispensable conditions provided by the specific organizational context.
- The second issue is related to clarifying if the Local (standard) protocol has to embrace multiple instances of *situated articulation* e.g. distributed in time, place, and/or organizational belonging. If so consider if/how a common protocol should be designed. Thus it is necessary to consider what implications a distributed/ heterogeneous user environment has for the need to customize the process protocol and/or the user interface.
- Reconciliation is always necessary but in case of a complex distributed and/or heterogeneous use environment it becomes more difficult. Thus it is very important to consider how to achieve the necessary reconciliation in the specific environment.

Sub-category 2.2 contains misfits related to internal and external data presentation; both reports and other kinds of summarized data used to provide overview for people belonging to a specific community and external partners interacting with this specific community. This

misfit category doesn't relate to the articulation of operational work processes in a direct sense, but it concerns providing overview of data and decision support for local (situated) stakeholders. The misfits found imply issues related to finding and/or interpreting local (situated) data in the database, and the Lack of flexible and quick ways to extract, manipulate and present (summarized) data. The implications of this category of misfits will be discussed in greater length under misfit category 4.

13.4 Misfit category 3: Implications for federal meta-work

This misfit category gives examples of situations where:

- Coordination is needed, but the coordination mechanism is inappropriate
- Workaround the local protocol in one department makes work in another department difficult (not able to follow the intended process protocol)

I find misfits caused by:

- Difficulties adapting the protocol complex to unique and indispensable requirements given by the organizational context e.g. specific use of service items (ex.3-3)
- Difficulties reconciling different stakeholders point of views (incomplete reconciliation) E.g. local/situated workarounds causing difficulties in other local/situated articulations (ex. 2-15)
- Difficulties understanding how to use coordination mechanisms as intended (conceptual schemes not understood by all involved parties)(ex. 2 – 2)
- Incomplete federal metawork (need for customization not realized or poor design of coordination mechanisms). (ex. 3 – 4 work-flow functionality)

Misfit category 4 provide examples of unrealized needs to customize the ERP protocol at metawork level, thus implying that federal as well as local metawork has been incomplete. It also provides examples of realized needs to customize the ERP protocol to adapt to the organizational needs, but these customizations have not been performed due to difficulties understanding how to customize the protocol without jeopardizing its ability to serve as a protocol. Finally category 4 provides examples of customizations performed that actually now makes it impossible to take advantage of other local (standard) protocols provided by the ERP package software.

Summing up misfits in category 4 implies that the implementation approach used caused difficulties realizing the need for customizations and provided insufficient insights how do perform customizations if a need was actually recognized. Furthermore this misfit category implies that reconciliation of different stakeholders' point of view has been incomplete during metawork.

Analyzing the misfits in category 4 I realized that different technical implementations of coordination mechanisms can be identified, and it turns out that the technical implementation has implications for the articulation process. The interplay between different technical implementations of coordination mechanisms and the nature of the articulation process is explained below.

In the case organization's ERP software four different ways of implementing coordination mechanisms can be found:

1. Simply sharing data (coordination achieved simply by getting access to data in the shared database)
2. Build into the application as (passive) status codes
3. Database triggers (looking out for an impulse and perform a step in the work process based on predefined criteria)
4. Using an actual Work-flow application on top of the basic ERP database and ERP application (also here some process protocols; work-flow templates are provided).

The four different types of coordination mechanisms require different combinations of federal metawork, local metawork and situated articulation. Moving forward they require increased metawork and at the same time they will decrease the flexibility for situated articulation. Below examples of the four different implementations of coordination mechanisms are given.

An example of coordinating by simply sharing data could be e.g. allowing a production manager to view the sales department's sales forecast and sales orders. By simply getting access to the data the production manager gets an input for both long term and short term production planning. The nature of the coordination mechanism require very limited metawork and has very limited implications for the local metawork and situated articulation. The value of the production plan may however increase if the two departments agree on the semantic of the data describing the sales forecast and sales order.

An example of coordination using codes/categories could be e.g. status codes for a purchase order related to the life cycles of the order (e.g. request for proposal, proposal accepted, delivery date confirmed by supplier, goods arrived, delivery accepted, order paid). Updating the status code can be interpreted as a signal that the order has been going through some processing and is now ready for the next step in the process protocol. In many cases different actors will be involved in processing different steps of the process protocol and the change in status code can be seen as a signal that someone else can take over. Deciding on the appropriate status codes to implement in the database require some anticipation about the work processes around purchase orders. If an application layer is added on top of the data model (data base) then different aspects of the anticipated work process have to be build into the code. The nature of the coordination mechanism depends on a variant degree of both metawork and situated articulation. The more detailed the process protocol the more the organizational articulation process depends on metawork.

If database triggers are used to implement segregation/coordination then the ERP system starts acting as an active participant in the anticipated work process. Thus the database will be looking out for a specific event to happen (an activation of a trigger), and when it does then a step in the process protocol will be performed based on predefined criteria. The nature of the coordination mechanism requires more of the anticipated work process to be build into ERP system. Thus the organizational articulation process depend more on metawork and less on situated articulation.

The fourth ways to implement coordination is using a work-flow tool on top of the ERP database and the traditional ERP application. If more advanced work-flow technology is implemented on top of the ERP database then the ERP system may actually become an active participant in the situated articulation. The work-flow technology will need a more or less complete picture of all the steps in the anticipated work process (in some cases it may actually be the only active participant "knowing" the complete process protocol), and the implemented work-flow to some extend take "control" coordinating actors in the anticipated work process; looking out for actors to perform their part of the process and making sure that the process is moved forward following the prescribed procedure. Thus in this case the organizational articulation process depends heavily on metawork and very little on situated articulation. Work-flow technology provide increased flexibility in regard to adapting (customizing) the ERP process protocol to the specific organization. However, at the same time the adapted process protocol gets less "tested" – having proven the ability to work in other settings is one of the arguments for implementing ERP software. Thus using work-flow technology the implementing organization may be required to take full responsibility for the metawork involved in developing a process protocol.

13.5 Misfit category 4: Need for taxonomic work as part of the articulation

Misfit category 4 contains misfits experienced by e.g. project managers, controllers and business managers relaying on data to perform their "management" work. The immediate implications of the misfits belonging to category 4 is a need for taxonomic work that goes far beyond what is necessary for local and federal metawork and at the same time the misfits point to defects in the factual properties of the software (a need for more flexible tools for finding, manipulating and presenting data). Thus at first glimpse misfit category 4 doesn't may not appear to be related to the articulation process, but it actually has implications for metawork as well as situated articulation and I will explain how below.

Being able to use data for financial/project control, planning and decision support require both data definitions and the data's semantic values to be uniform; it requires *standard data* which are comparable over time and across instances. A careful designed ERP database is able to provide the common data definitions (at a technical level) and the database management tool may be able to ensure that values entered are within certain limits. But uniform (comparable) semantic values depend on all parties involved developing a shared understanding how the data is used and should be understood. In the case organization for example project managers make approximate and uncommitted bookings of resources in order to keep their options open, while department managers (responsible for the resources) need the bookings to be exact in order to make resource planning easier and ensure a good utilization of the resources. At a different level in the organization management use the same data to plane the influx and departure of employees. Thus the same piece of data is used by different parties for operational purposes, and planning and decision support at different levels. Some kind of agreement how to understand and use the specific piece of data has to be reached. Most data used for planning and decision support originate in operational data (data entered and/or used as part of the situated articulation), thus indirectly the taxonomic work gets entangled in the articulation process as it depends on the articulation process to provide well understood (enforceable) conceptual schemes for data entrance. Taxonomic work is in itself complicated,

but the fact that situated articulation needs to resolve conflicting, unforeseen and/or changing situated conditions makes it even more complicated.

Chapter 14
A theory for an ERP articulation process

In this chapter the complete theory of ERP implementations understood as an ERP articulation process is presented.

The first section (14.1) explains how ERP implementations can be seen as an articulation process. Then in section 14.2 I specify how customizations to the process protocol and the IT artifact can take place. This is finally followed by section 14.3 where I discuss reconciliation when implementing ERP

14.1 Understanding ERP implementations as an articulation process

In this section I refine the theory understanding ERP implementations as an articulation process; *an ERP articulation process* where the insights provided by the analysis of the misfit categories are incorporated.

Using the terminology developed in chapter 12 implementing ERP systems can be understood as two entangled tasks:

1. Conducting an articulating process developing a *federal process protocol* (also includes the local process protocols) by adapting a pre-defined complex of organizational processes (an *ERP process protocol* provided by the ERP software) to the federal context, and
2. Constructing an IT artifact that reflects the federal process protocol, and at the same time provides a user interface that supports situated articulation. The user interface can be understood as templates or other kind of guidance for end-users in order to make it easy for them to provide coordination in the way it is anticipated by the process protocol, and at the same time provide data of uniform semantic value for management processes at all levels

These two tasks have to be conducted in a way that takes the underlying rationale behind standard software into consideration. Thus the implementation has to consider a desire to:

- Rationalize (do more with the same resources) organizational processes using standardization and segregation as the driving force
- Achieve coordination across a distributed and heterogeneous environment using standardization and segregation as the underlying philosophy
- Provide standardized data (comparable over time and across instances) supporting management processes at all levels
- Reduce (minimize) the cost of ownership of ERP software

The ERP articulation process can be understood as having two different but still related perspectives: a *coordination perspective* and a *semantic perspective*. The coordination

perspective focuses on articulating operational work processes emphasizing the design and use of coordination mechanisms as a way to tie work processes together in a complex of processes; a *federal process protocol* (explained in section 12.5). The semantic perspective focuses on management processes (financial control, operational planning and decision support) emphasizing the need to provide *standardized data* (explained in section 13.5).

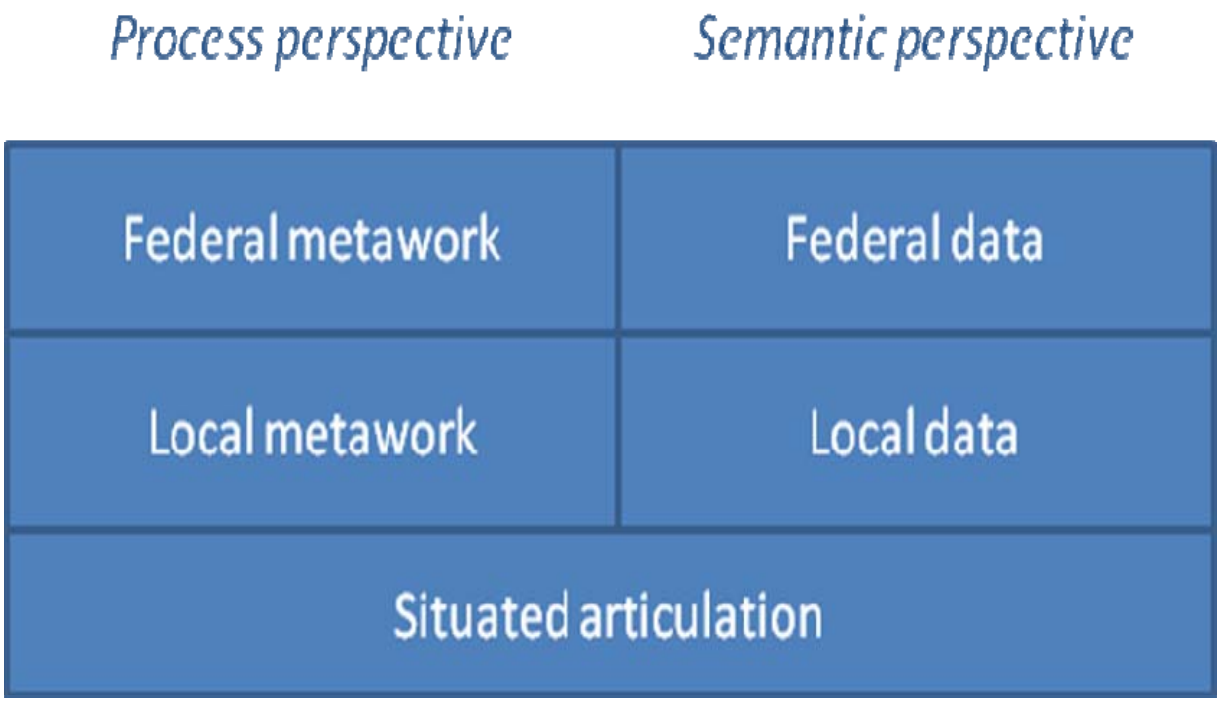


Figure 14.1: My theory with three layers and two perspectives

Theoretically an articulation process can be conducted without considering the factual properties of the IT artifact intended to provide the bearing structure of the process protocol and support the situated articulation. But when implementing ERP package software the articulation process and the construction of the IT artifact cannot be separated, and the IT artifact influences and constrains the articulation process in different ways.

The standard ERP package can be considered a traditional IT artifact in the sense that it includes a database, application logic, screens providing the user interface etc. But it is also special in the sense that it holds a predefined and build-in abstract process protocol. Thus somebody outside the implementing organization has developed an abstract (global) model of a *federal process protocol*; a model that is believed to be of relevance to the implementing organization. Furthermore the process protocol and the IT artifact are reciprocally binding. Due to these properties the ERP package software is supposed to replace metawork (if the model is complete) or at least make metawork much easier (only customizations need to be considered); therefore ERP package software is expected to reduce the cost of (constructing) the IT artifact, and at same time ensure that the process protocol are aligned with the IT artifact.

My case study at Alfa shows that the predefined organizational process model (the ERP process protocol) build-into the ERP software was not complete, hence all layers in the articulation process needs to be included in the implementation process. During both federal and local metawork the ERP protocol becomes a stakeholder in the articulation process. The ERP protocol rather than the organizational context serve as the point of departure for the articulation process, and customizing the ERP protocol has to be argued carefully because it will increase the demand for metawork and disrupt the reciprocally nature of the process protocol and the IT artifact. Thus customizing the ERP process protocol may in itself be complex because derived effects of a customization has to be identified and a new articulation has to be negotiated; changing a local protocol could make it impossible to use a coordination mechanism as intended, or it may change the semantic value of data. Such changes has to be articulated in a way that ensure completeness of the metawork. But customizing the process protocol could have more derived effects as they may disrupt the reciprocally nature of the process protocol and the IT artifact. If that is the case then the derived effects for the IT artifact has to be investigated and the changes to the IT artifact needs to be analyzed, specified, implemented, and tested.

The ERP software is also involved in the situated articulation. First it may help the user understand how to perform situated articulation that is in accordance with the federal process protocol. This can be done in two ways: (1) As a bearing structure of the protocol the software may provide guidance (to some extent enforce) how to perform the situated steps in the process protocol and how to "use" coordination mechanisms as intended by the federal process protocol, and (2) a well designed user interface may provide guidance how to produce standardized data. If database triggers or more advanced work-flow technology are used as part of the ERP implementation then the ERP software actually becomes an active participant in the situated articulation thus performing steps in the processes or maybe actually being in control of the workflow (as explained in further details in chapter 13 section 13.4 above).

As especially misfit category 2 implies (se section 13.3 for more details) another complication related to the ERP articulation process is that the notion of situated/local can become very complex when implementing ERP software in heterogeneous and distributed environments. Thus the same local protocol (process procedure as well as code in the software) may be intended for many different use situations. E.g. in the Alfa case organization the ERP process protocol and the human computer interface for project management are intended for projects with very different characteristics and users groups with very different backgrounds (conceptual schemes). Thus processes that are local in relation to the ERP protocol (the ERP software) may need to be adapted to many different use situations. Thus the same local protocol may be involved in multiple instances of situated articulation. If accommodating a need to support many different situations at the same time the design of the local protocol may become very complex, alternatively it will make situated articulation very troublesome. You may suggest developing a customized version of the local protocol (a situated protocol) for all the different use situations to get around the issue, but as rationalization (provided by segregation and standardization) is both the goal and the mean when developing federal process protocols then deciding on customizing local protocols may not be an easy choice.

The ERP articulation process may also be complicated by the fact that it often is impossible to rely on a common authority with the power to adjudicate differences between stakeholders or

command them to do things in a specific way. Thus in order to make things work (and have solutions accepted) it is often necessary to persuade rather than command the stakeholders. If sufficient reconciliation is not performed during federal metawork the battle between stakeholders may continue into situated articulation and the demand for customizations may persist. Hence reconciliation during the organizational adaption of a global protocol is challenging but necessary.

14.2 Specifying customizations to the process protocol and the IT artifact

Focusing on specifying the customizations needed to adapt/transform the ERP protocol into the federal process protocol and the (ERP software) IT-artifact to the (federal) IT-artifact the first step is to understand when/where the existing/desired organizational processes are differing from those suggested by the ERP protocol; thus gaps needs to be identified, and if a need for customization is found then the next step is to decide how to implement the customization.

When performing the process *gap analysis* it is necessary to provide an understanding about the two protocols that needs to be compared. Hence someone from the organization (or someone/something endorsed by the organization) needs to provide "knowledge" about (a specification of) the organizational processes (as-is and intended to-be), and someone or something needs to provide "knowledge" about (a specification of) the ERP protocol. Furthermore when conducting the gap analysis the "knowledge" provided about both protocols needs to be in a format that allows comparison.

One of the complications of performing the gap analysis is that it is necessary to understand both the local protocols and the coordination mechanisms that ties local protocols together into becoming federal process protocols. Hence it has to be clarified how to split the process complex into local protocols in order to investigate if the local ERP protocols fit the local organizational protocols. Because a "local" process protocol may be involved in multiple and/or distributed *situated articulations* it could be rather complex to investigate if the local protocol will be able to accommodate situated articulation throughout the organization.

Another aspect of the analysis is to specify which coordination mechanisms the local protocol share with other local protocols and then investigate if a shared understanding how to use the coordination mechanism is/can be established.

Finally, the taxonomic perspective of the articulation process has to be addressed. Thus it is necessary to understand if local and federal protocols will be able to provide standardized data for management processes on all levels.

The results of the gap analysis provide a point of departure for investigating how to perform customizations in order to bridge the gaps. Deciding how to bridge the gaps means that the articulation process is in *design mode*, an overview of the activities involved for the articulation process in design mode is provided in figure 14.2 below. In design mode both federal and local articulation needs to take place. Customizing the ERP protocol is one option, adapting the organizational processes protocol to the ERP protocol (asking the organization the change the way it operates) is another option. The two options may be used in a

combination. When deciding on customizing the ERP protocol it is important to be aware of the entangled nature of the ERP protocol and the IT artifact providing the bearing structure of the ERP protocol. Hence both deep knowledge about the technical implementation of the ERP protocol and knowledge how to implement the changes in the protocol is needed in order to specify the customization.

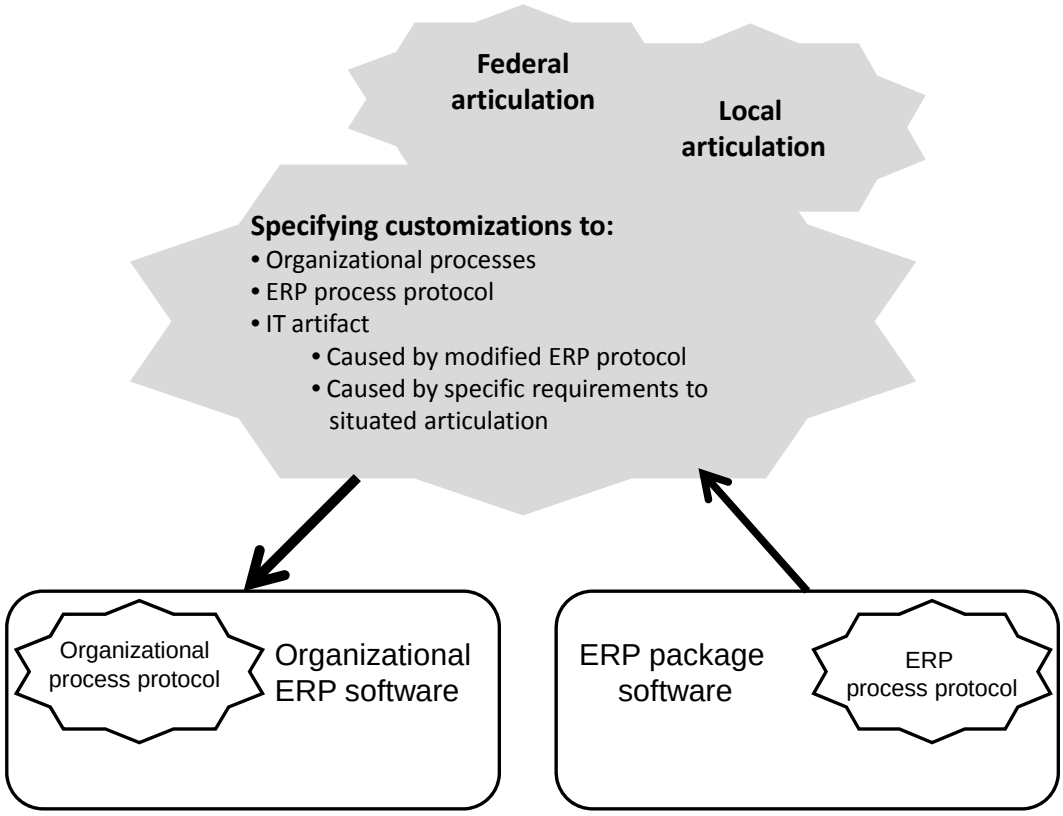


Figure 14.2: ERP articulation process in design mode

When looking at the IT-artifact as the design object then the customizations to the ERP protocol only provide part of the picture. Apart from carrying the bearing structure of the federal process protocol the IT artifact also plays an important role during situated articulation providing guidance for situated users how to perform a step (the next step) in the process protocol, and it can support/enforce users to provide (semantically) standardized data.

Analyzing if it is necessary to improve the guidance for situated articulation holds two parts; (1) it require you to understand the properties of the new federal process protocol; the steps in the situated part of the local protocol, how to perform the steps in order to provide the intended coordination, and understanding the semantic value of standardized data. (2) You need to understand the context of situated articulation; the physical environment, the users' skills, management preferences etc.. Thus in the analysis you need to compare the local protocol to the organizational context; first you need to realize if conceptual schemes are out of line, and then the next step is to decide how to improve the situation; customizing the IT software to provide better guidance for the situated articulation, or improving the situated users understanding of the conceptual schemes the protocol are based on. The final result of

the analyses is specifying customizations caused by specific requirements to situated articulation.

14.3 Reconciliation when implementing ERP

ERP implementations may take place in very heterogeneous and distributed environments. In some areas of the implementation there may be a common authority or another arrangement that can adjudicate differences between stakeholders, but in other areas of the implementation that may not be the case. In the later case differences among stakeholders have to be resolved arriving at some kind of collective decision or policy that is reasonably efficient, effective, and equitable. If the federal protocol isn't used as intended then the rationalizing and coordinating effect of the protocol will in most cases fail. And if local protocols aren't accommodating the situated articulation then they may not be used as expected/intended. Hence, to improve the chances of making the federal protocol and local protocols successful it is necessary to better understand the areas that need reconciliation and how to perform it.

If perceiving the ERP implementation as an articulation process some obvious conflict areas can be identified, thus it provides an understanding for areas where reconciliation is needed.

***1) Reconciling how (much) the ERP process protocol are allowed to constrain the design of the federal process protocol*:** Deciding on the degree of metawork allowed/necessary transforming (adapting) the ERP protocol into a federal protocol may require reconciliation. Put in a different way different stakeholders may try to influence if (to what extent) the ERP protocol is allowed to constrain the design space of the federal protocol and local protocols. If all powerful stakeholders perceive the ERP protocol to be a complete and desired federal protocol then very little design space may be necessary/allowed, hence little metawork is anticipated. If on the other hand the ERP protocol is perceived by influential stakeholders to be just a starting point then a larger design space may be necessary/allowed, hence much metawork has to be conducted.

2) Reconciling different stakeholders' needs and point of views during the design of the federal process protocol: No matter how much design space is allowed (how much the ERP protocol are allowed to constrain the design of the federal protocol) there will always be some space available. When exploiting the allowed design space and deciding on solutions it is important to understand how to balance the needs and preferences of different groups. Hence reconciliation is necessary; it is necessary to support a negotiation process arriving at some kind of agreed/committed design of the federal protocol.

3) Reconciling different situated stakeholders needs and point of views during the design of local protocols: It is also necessary to reconcile the needs and preferences of different situated stakeholders when exploding the design space allowed for the local protocol (local metawork); understand how the needs of all users of a local protocol can be clarified and negotiated. Many geographically distributed groups with the same profession may be expected to use the same local protocol. How can their needs and preferences be reconciled? A local protocol may also be intended for heterogeneous groups of professionals. E.g. for a purchase protocol used throughout the organization by professional purchasing employees as well as employees using it sporadicly to buy paper and pencils. Or another example may be a

project management protocol used for different kind of projects (large construction projects, maintenance projects, internal IT projects, external projects in the graphical department…..). How can the needs and preferences of heterogeneous groups of users/types of the same phenomenon be reconciled? Finally a "local" protocol may also be intended to coordinate distributed groups (users) covering different aspects of the same local protocol. Thus there is also a need to reconcile coordination mechanisms within local protocols.

4) Reconciling federal stakeholders' needs and point of views when specifying data definitions and semantic values of data in order to obtain standardized data for management processes: The need for standardized data (supporting management processes on all levels) may trigger conflicts between different groups within the organization because specific parts of a process may attract more/less attention than usually (indirectly changing the groups power or the importance of their work) or because it makes it possible to compare individuals or groups within the federation performing the same tasks.

On top of that the situated users may be required to work in a very specific (standardized) way in order to obtain standardized data. Thus situated stakeholders may not experience any benefits related to their own work from a standardized work process, but it may be necessary in order to obtain standardized data. Hence there is a need to reconcile different stakeholders' points of views in order to make them all understand (agree) why a specific semantic value of data is needed, and how it is used.

Summing up; an ERP articulation process is performed in order to arrive at a reconciled federal process protocol, and an ERP articulation process can be understood to have two different but still related perspectives: a *coordination perspective* and a *semantic perspective*.

The coordination perspective focuses on articulating operational work processes emphasizing the design and use of coordination mechanisms as a way to tie work processes together in a complex of processes; a *federal process protocol*.

The semantic perspective focuses on management processes (financial control, operational planning and decision support) emphasizing the need to provide *standardized data*. Both perspectives can be understood to have three layers: A *federal articulation layer* (focus on coordinating between local protocols), a *local articulation* layer (focus on developing local protocols for specific functions in the organization) and a *situated articulation* (actors actually performing the processes covered by the protocols) layer.

Part V - The fourth learning cycle

Part V - *The fourth learning cycle* contains three chapters.

First – again – a short research method chapter – no. 15 – where I discuss how to apply the theory for my data.

Then in chapter 16 I re-interpret the empirical findings provided by the case study focusing on how design artifacts are used to support the different layers and perspectives in an ERP articulation process.

And then in chapter 17 I re-interpret the empirical findings provided by the focus group study focusing on how metawork performed during the initial design process is perceived by the ERP professionals.

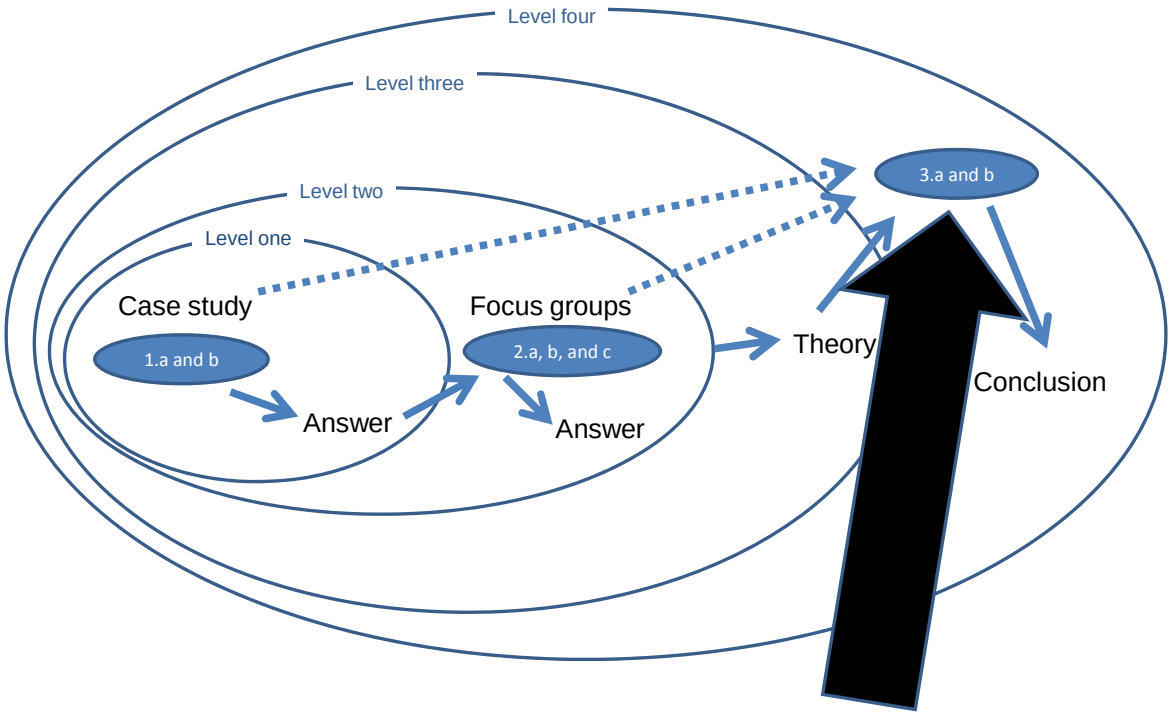


Figure V.1: The overall structure of the Ph.D. thesis with a thick arrow indicating where part V belongs

Chapter 15
Research Method for the fourth learning cycle

In the fourth learning cycle I apply the ERP articulation theory (developed in learning cycle three) to my empirical field material. Below I explain in more details the approach used answering the two research questions belonging to this learning cycle.

15.1 Research design for the first research question in the forth learning cycle

The misfit analysis in chapter 5 left an impression of an ERP implementation with many and serious design issues, and in chapter 6 the analysis of the knowledge integration capabilities imply that insufficient knowledge for design was developed during the very important initial set-up of the new ERP system (phase 2 in the implementation). In this fourth and last learning cycle I use the theory developed in chapter 14 to analyze the design process performed in the case organization. The research question formulated for this analysis is:

> How is the ERP articulation process approached in Alfa's case?

Thus I reinterpret my empirical material using an understanding of ERP implementations as an articulation process, and design as a cooperative and mediated activity. Doing so provides new insights into the implementation process. The analysis focus on the second phase in Alfa's ERP implementation, this phase could be understood as focusing on articulation work in design mode.

The analysis is based on an understanding of design as a mediated process. Design can be perceived as a cooperative enterprise "*where different people with different professional backgrounds and different motives are engaged in creating something new, the object of design"* ((Bertelsen 2001), p. 18). This notion of design fits well with both design objects in ERP implementations and the large amount of stakeholders that seems to be involved. The stakeholders involved have significant differences in professional backgrounds and organizational belonging, but still they are expected to design the new federal process protocol and the federal ERP software together. Hence there appear to be support for perceiving ERP implementations as a cooperative design enterprise.

Following this idea of design as a cooperative enterprise then the *design activity* can be understood as an activity mediated by design artifacts. *"Design artifacts mediate system development as cooperation, both as explicit means of cooperation (e.g. status reports and specifications) and as means for the sharing of experiences, insights, and visions about the design object* ((Bertelsen 2001), p. 18)". If using Star´s (1989) terminology design artifacts can be understood to serve as boundary objects that adapt to different situations of application while maintain their identity, and thereby they are able to mediate divergent needs and viewpoints in the cooperative process.

As explained in section 12.5 an articulation process can be understood to have three articulation layers with no given order or hierarchal importance. But if articulation is involved

in an ERP implementation, and design is understood to be a cooperative process of transforming artifacts from the domain of use, then part of the (ERP) articulation process has to take place in a zone of design. Thus abstract representations of situated articulation has to be broad into the design process so that talking about the processes and considering how to change the process protocol can take place, and in this zone of design both federal metawork and local metawork is performed.

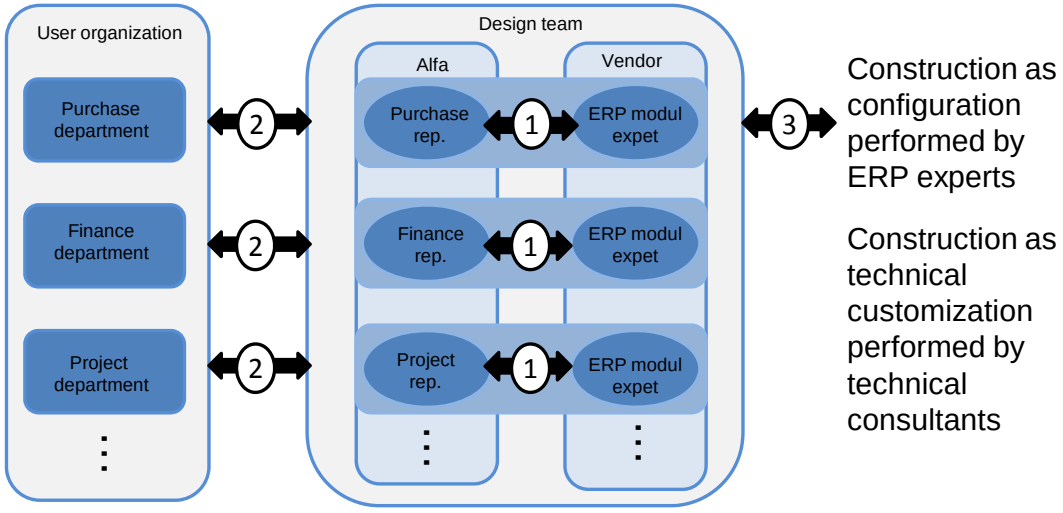
In general different kind of design artifacts can be involved in order to mediate a design activity; programming languages, case-tools, specification standards, developing methods (Bertelsen 2001). This will also apply to ERP implementations, but I am especially interested in understanding which design artifacts are used to mediate between the parties involved in the cooperative design process aimed at developing a federal process protocol and an IT artifact holding the bearing structure of the process protocol and templates (guidelines) for the situated articulated.

Design artifacts can be understood to mediate between use and design, between users and designers, and between representations and construction (Bertelsen 2001). But to better accommodate ERP implementations understood as an articulation process I have chosen to split the original three areas design artifacts are expected to mediate into six areas. For the first two areas a split allow me to distinguish between mediating local metawork and federal metawork. For the third area a split allow me to distinguish between mediating for construction as configuration and construction as customization. Thus the six areas I am interested in are:

1. Mediate between “users and designers”
    a. Between organizational representatives and ERP experts when performing local metawork as a design activity
    b. Between organizational representatives and ERP experts when performing federal metawork as a design activity
2. Mediate between “use and design”
    a. Between use and design/design and use of local protocols and standardized data
    b. Between use and design/design and use of the federal protocol and standardized data
3. Mediate between “representations and construction”
    a. Between representations and construction of the IT artifact as configuration
    b. Between representations and construction of the IT artifact as technical customization

In figure 15.1 the primary groups involved in Alfa’s ERP implementation is depicted. The groups belong to either the *user organization* or the *design team*, and areas where shared insights are needed are illustrated with black arrows. The numbers on the arrows correspond to the three areas that need to be mediated by design artifacts (just explained above). The design team consists of several functional sub-groups and equivalent functional groups can be found in the user organization.

The result of the analysis is described in provided in chapter 16.



1. Mediate between o*rganizational representatives and ERP experts*
2. Mediate between *use and design*
3. Mediate between representations and construction

Figure 15.1: Primary groups involved in Alfa's ERP implementation. Areas where shared insights are needed are illustrated with black arrows.

15.2 Research design for the second research question in the forth learning cycle

The second research question for this learning cycle is:

> How does the ERP professionals' perception of ERP implementations correspond to ERP articulation work?

To answer this question I re-interpret my empirical material from the focus group study. As a starting point I take the metaphors identified in section 9.7.

The metaphors have different perceptions of who is conduction process design in an ERP implementation and it is this dimension in the metaphors I explore.

The result is a two-by-two framework presented in chapter 17. The framework shows four different perceptions of how to conduct metawork; the perceptions differ in two dimensions: (1) how much explicit metawork is anticipated, and (2) how much the ERP software is expected to constrain the design space.

Chapter 16
Re-interpreting the Alfa case

The aim of this chapter is to provide an understanding of how Alfa's ERP articulation process is approached. Thus the design artifacts used (identified as used) for design activities and the experiences using them in Alfa's ERP implementation is analyzed in order to understand how knowledge for the gap analysis is provided, how knowledge for deciding on how to bridge the gaps found is provided, and how reconciliation is handled.

I have chosen not to include the overall implementation method and artifacts related to project management. Instead I have focused on the primary design artifacts thatare used to mediate the ERP articulation process. The research approach used and the theory of design artifacts are presented in chapter 15. The analysis conclude that the design artifacts used in Alfa's ERP implementations and the way they were used lacked the ability to mediate the articulation process in phase 2, hence complicating and prolonging the organizational articulation process.

The remainder of this chapter is structured as follows. First I analyze the mediation of local metawork in section 16.1. Then federal metawork and the mediation hereof are analyzed in section 16.2. Section 16.3 then looks at mediation between use and design in local metawork, and 16.4 does the same for federal metawork. In section 16.5 the mediation between representation and construction is then analyzed. I then sum up the use of design artifacts in section 16.6. And then, finally, I conclude the chapter in section 16.7.

16.1 Mediating local metawork

In Alfa's design team local metawork was performed within four functional sub-groups; a purchase groups, a finance group, a project management group, and a resource management group. Each functional sub-group consisted of a number of user representatives from the specific functional area, and one or more ERP consultants with knowledge about the local processes within the ERP package (typically collected in a specific module in the software).

Figure 16.1 illustrates that different sub-groups existed within the design team. Each functional sub-groups are focusing on a sub-set of processes within the ERP protocol; a local process protocol. The black arrows illustrate that shared insights are needed between the organizational representatives and the ERP experts within each sub-group.

The user representatives are expected to provide knowledge about the existing local/situated processes. The ERP experts are expected to provide knowledge on how to use the configuration parameters for different process options and provide knowledge about the alternative local process protocols as they are anticipated/prescribed by the ERP protocol. In the functional sub-groups the participants from both the case organization and the vendor's organization in most cases had a common background (education or practical experience) related to a specific profession, thus to some extent they have a shared vocabulary and shared theories about the work related to a specific local protocol.

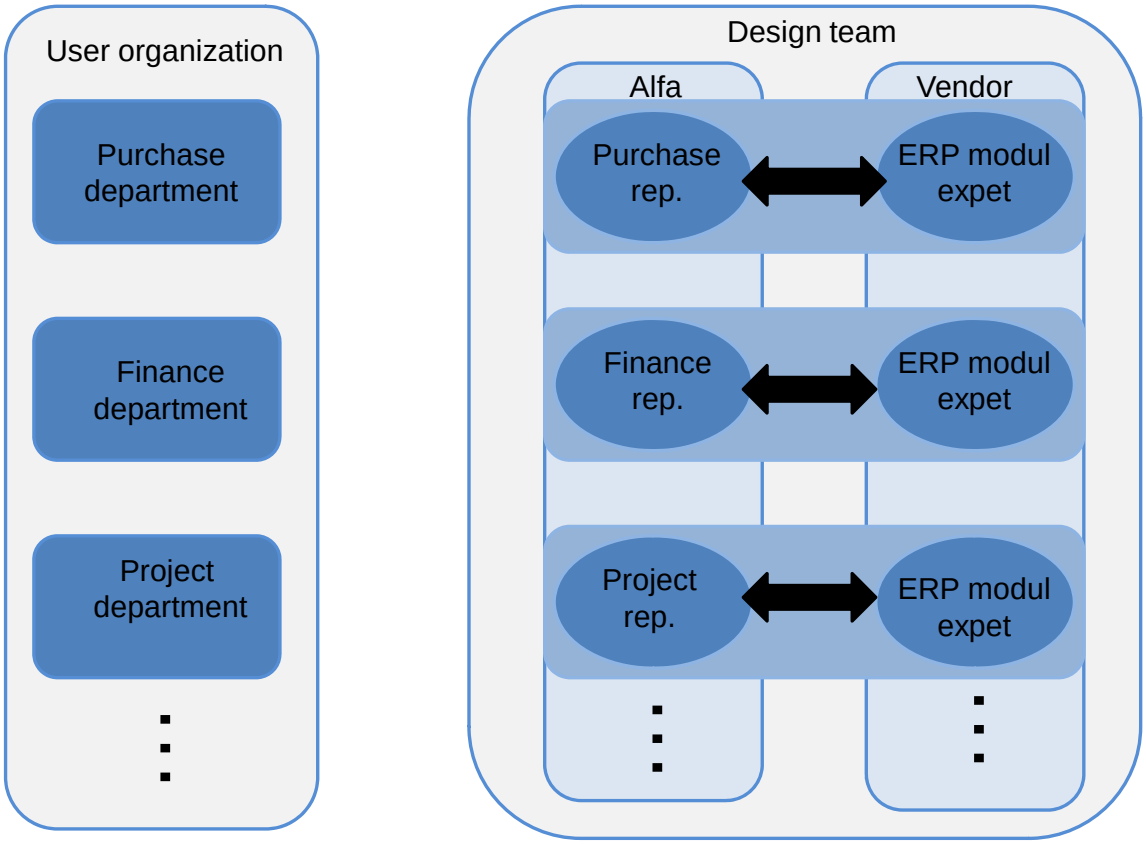


Figure 16.1: Sub-groups within the design team. Arrows indicate need for shard insight

Ad-hoc artifacts may have been developed within the functional sub-groups e.g. drawings on a blackboard, but I have not been given examples of such ad-hoc artifacts due to the fact than none seems to have been preserved. Thus the primary sustainable design artifacts used within the functional sub-groups are: Process diagrams, workshops, the ERP software applied as a prototype, and the requirements specification. In the text below each design artifact is considered in more details.

The requirements specification developed prior to selecting the ERP software and partly revised and detailed during scoping was used as a design artifact. As a design artifact it served to mark the boundary of the implementation, and focused the ERP experts' attention on areas that was specified. It did not provide much insight and wasn't used actively during the design activities but mainly served as a checklist for the ERP experts.

Process diagrams composed by the ERP software vendor and provided as part of the implementation "tool-box" were used by the functional sub-groups. These process diagrams can be understood as representations of the business processes developed by people outside the actual design team. At the point of departure the process diagrams are nothing but something very abstract with limited ability to depict the actual implementation of the process in the ERP software (the ERP process protocol) or depict the actual situated articulation that takes place in the organization.

Using the process diagrams to mediate properties of the ERP process protocol to user representatives lead the user representatives to complain that a false perception of understanding was induced. The abstraction level was too high thus the user representatives only develop a weak understanding of the ERP process protocol and the software's capabilities. Furthermore it was possible to model something that could not be accommodated by the ERP software.

Using the process diagrams to mediate properties of the existing/to-be local process protocol the ERP experts complain about the level of details provided by the process diagrams. Thus when used in cooperation with the user representatives the process diagrams did not help facilitate discussing or documenting the necessary details about the user organization in order for the ERP experts to construct the ERP system in a way that reflect the user organizations requirements (needs).

Hence using the process diagrams as a common representation of properties of the federal process protocol to be, the process diagrams seem to make it too easy for user representatives and ERP experts to arriving at a common (partly false) perception of the "fit" between the organizational practice (the existing situated articulation) and the properties of the software. The lack of details seems to be a problem performing local articulation in the functional sub-groups. Especially because the technical implementation of the local process protocols in the software reflects a very specific interpretation of the process diagrams.

Several workshops took place e.g. the conference room pilot and configuration workshops. These workshops were mainly used for exchanging information. The user representatives were interviewed about the work performed in the organization, and based on this information the ERP experts would set up the system and show it to the user representatives. This would often be followed by a discussion of why the design didn't fit Alfa's way of working, where after the ERP experts would try very hard to explain/convince the user representatives why this way of working would be a useful/a possible way for Alfa to work.

The ERP software applied as a prototype was used as a design artifact within the functional sub-groups. A preliminary (pre-configured) version of the ERP system was used in "configuration" workshops in the functional groups, and it helped the user representatives get some experience with the future system. Based on these insights the changes necessary was discussed, and if possible the configuration parameters was changed. Thus the IT artifact served both as a continuously moving object of the design activity and at the same time as a design artifact mediating the creation of insights and visions into the local ERP processes protocols and the properties of the IT artifact.

The emerging "prototype" of the ERP software had some ability to mediate insight building within the functional sub-groups about the IT artifact; many discussions within the group were initiated providing some mutual understanding and transferring insights from the use domain/user representatives to the ERP experts/the design object and vice versa. In most cases the discussions and the decisions taken during discussions was only documented in the form of configuration parameters; changing the properties of the prototype/design object. If a need to customize the software was identified then a request for change was performed. Although using the software as a prototype and carry out discussions did provide shared insights, the way they were performed clearly had a focus on the IT artifact and the ERP

experts' need to understand specific details about the use domain in order to configure the software. Thus the user representatives' need to identify the gaps between the ERP protocol and the existing local protocols had very limited attention, and the way the prototype was used it provided no or a very superficial understanding of where changes in the local protocols/situated articulation could/should be anticipated. Hence it was left entirely to the individual user representatives to consider if and how the new process protocol would require changes in the situated articulation, and in case a need for change was identified whether it would be acceptable for the situated stakeholders.

Looking at the combined use of the process diagrams and the emerging prototype they can be perceived as providing a top-down and a bottom-up approach respectively. The process diagrams provide a high level (abstract) representation of the federal protocol, and the prototype a very complex and detailed representation. However no "bridge" existed that ensured consistence between the two models, thus they cannot be expected to provide a complete and comprehensive picture.

16.2 Mediating federal metawork and taxonomic work

Federal metawork is important in order to provide well-designed coordination mechanisms that allow local protocols to be coordinated as frictionless as possible. Thus it is necessary to identify the need for integration/coordination that cut across local protocols, and consider the design of the coordination mechanisms provided by the ERP protocol are appropriate for Alfa's specific context.

Support for insight-building related to federal metawork and taxonomic work requires knowledge that goes across the functional sub-groups. This includes:

1. Identify gaps between the ERP protocol and a federal process protocol that will accommodate the specific organizational context
2. Specify customizations to the organizations' existing processes in order to adapt to the ERP protocol
3. Decide on the design of coordination mechanisms that are shared by two or more local protocols, and deciding on the technical implementation of the coordination mechanisms
4. Identify the need for standardized data; identifying management processes and specifying customizations to the user interface of the IT artifact in order to provide the desired guidance for obtaining standardized data
5. Specify technical customizations to the local ERP protocol and/or the IT artifact if necessary

In Alfa's project organization federal metawork and taxonomic work will require some shared insight building between functional sub-groups. In figure 16.2 the functional sub-groups within Alfa's design team are illustrated and the black arrows illustrate where they need to build shared insights.

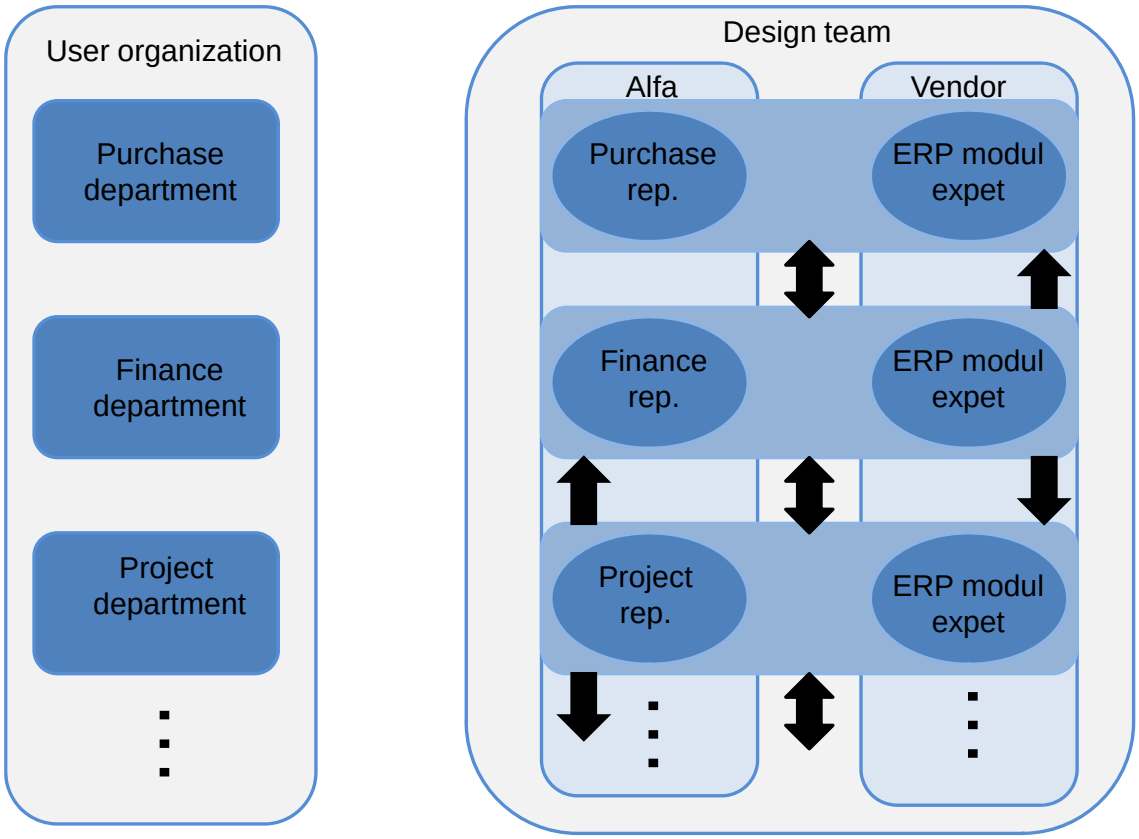


Figure 16.2: Functional sub-groups in Alfa. Arrows indicate need for shared insight

In Alfa's implementation very limited traces of federal metawork can be found. In some of my interviews with user representatives they refer to the project manager and an ERP architect when asked about how issues related to federal metawork was covered. The vendors architect and the ERP project manager did meet now and then to discuss design issues cutting across functional groups. However they had no formalized way of sharing their insights with the functional sub-groups and the functional groups had no formalized way of providing insights about the actual design of local protocols designed in the functional sub-groups. Several interviewees tell about a very late workshop involving all functional sub-groups at the same time. This workshop discovered serious coordination issues and unresolved issues that all the involved parties believed someone else took care of.

Neither the process diagrams nor the prototype (the emerging configuration of the ERP software) were used systematically to mediate insight building between functional sub-groups. Each sub-group focused on their own module/part of processes, and spend almost all their project time working within their own functional area. At the rare occasion they meet to discuss the design, differences in professional backgrounds and lack of knowledge about the other modules in the ERP software constrained the insight building. As the project progressed a few of the user representatives got involved in more than one sub-group. Thus they started to develop cross functional knowledge. This made them able to see relations between functionality (work processes) in different sub-groups (different software modules). As it happened these new insights came about by coincidence and they made it possible for user representatives to bring up federal design issues that otherwise would have been overlooked.

Writing test cases and performing the test could have been an opportunity to address design issues cutting across functional sub-groups. But again the sub-groups were mainly working individually. The test cases were developed with a functional sub-group perspective and most of the testing was performed by people related to a specific sub-group. Only very late in the test process did the testers get together in a room to perform specific test sequences that provided insight into how the system performed cross-functional coordination.

16.3 Mediating between use and design when performing local metawork

When performing local metawork mediating between use and design can be seen as situated articulation during local metawork. This have two purposes; first investigating whether conceptual alignment between the suggested local protocol and the situated users is in place, and second to ensure reconciliation among stakeholders.

As discussed in chapter 5 it wasn't always possible to determine with absolute certainty if a misfit was caused by lack of knowledge about how to operate the ERP software or if the case organization's context left no other possibility than to customize the software. But all the interviewees acknowledged that both issues played an important role. Thus for the functional sub-groups and their peers and managers insight building required two flows of information sharing with opposite directions; insights into the existing practice needed to flow from the user organization to the functional sub-group, and insights into the suggested/decided design needed to flow from the functional sub-groups to the peers and managers in the organization.

In figure 16.3 the groups that are involved as either "provider" or "receiver" of the information is shown. A reference group for each functional group was formed with representatives from the user organization, and for all processes a process owner was appointed. How the reference group and process owners was included in the process.

Mediating between use and design can be perceived as actually having two directions; providing insights about use (situated articulation) to the design, and the other way providing insights about the design (the local process protocol) to the stakeholders involved in situated articulation. In an ERP implementation the first direction (informing design about the present situated articulation) could be perceived as insight building for design (configuration and customization), and the second direction as insight building in order to understand the derived implications of a specific design suggestion for situated use.

In Alfa's implementation a few representations can be identified that can have been used or actually was used to mediate between use and design: process diagrams, business goals for the individual functional sub-groups and a requirement specification. All three are artifacts developed prior to choosing the actual ERP package. No new representations of the previous situated articulation were developed during phase 2. To mediate between design and use a process manual was developed at the very end of phase 2.

In the case organization a vision about the future ERP system is developed during phase 1 of the ERP implementation that involves all of the functional sub-groups. The vision is documented by high level process diagrams, requirements specification, and business goals for the individual functional sub-groups. Thus during phase 1 three important artifacts are

constructed within each of the functional groups facilitating the development of shared insight about the current practice and the vision for the future. The only artifacts carrying over to phase 2 (design of local and federal protocols and the IT artifact) is however the requirements specification and the business goals. Both artifacts are rather abstract and say very little about the existing practice.

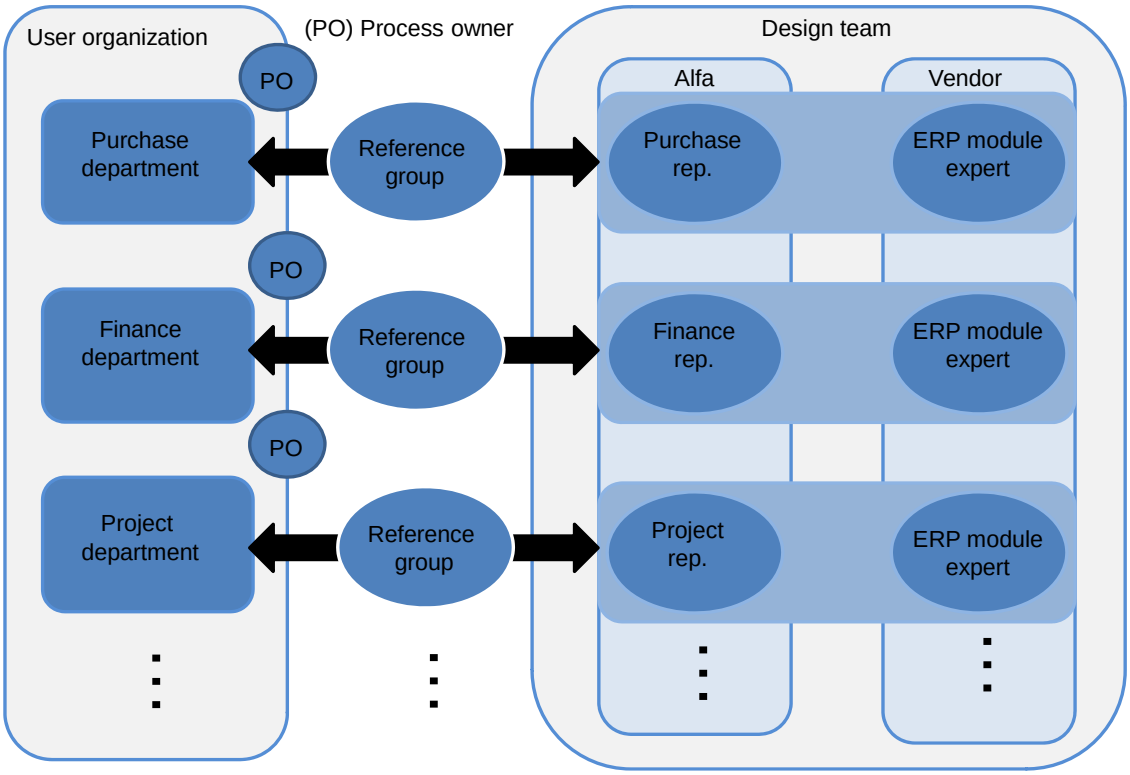


Figure 16.3: Groups involved as either “provider” or “receiver” of information

The transition to phase 2 in the ERP implementation mark a change concerning who is involved in the ERP implementation. Many of the people directly involved in developing the process diagrams etc. in phase 1 is also included in the design team although there are some changes. The ERP design team however also includes external ERP experts from the chosen vendor. These new people have no knowledge about the process performed in phase 1. Thus they miss the shared insights developed during phase 1.

The requirements specification developed in phase 1 is kept as an important document, it is the basis for the contract with the vendor (and Alfa’s organization) and it is used as a check-list when reporting progress and discussing change requests. In the cooperation between user representatives and ERP experts in the sub-groups considerable interpretation of the requirements specification is necessary. Furthermore it is quickly realized that both the development of the requirements specification and the ERP vendors’ interpretation of the requirements specification had been based on assumptions that do not hold. Thus the user representatives realized that requirements had to be added because unrecognized assumptions

about the ERP package software turned out to be false. On the other side vendor consultants realized that when answering the request for proposal the interpretation of the requirements specification had been based on assumptions about the organizational context that didn't hold. Hence dual broken expectations resulted in new requirements and new change requests. However no new representations of the organizational practice; the existing situated articulation is developed. Details about the current practice are exclusively provided by user representatives in workshop discussions with ERP experts and by giving feedback on the ERP software serving as a prototype.

Alfa's project organization provided some overlap of persons between the functional groups in Alfa's user organization and participants in the functional sub-groups of the design team. Some user representatives worked full time and others part time in the ERP design team. It was however still difficult to ensure that all local stakeholders was represented in the design team. Hence an ERP reference group for each functional sub-group was established to provide a forum where a better representation of situated stakeholders was possible. Each reference group represented situated stakeholders related to the local protocols designed within the functional sub-group, and the reference group was meant as a forum where user representatives could discuss design issues.

The members of a reference group were not expected to participating in actual design work, but should just provide an easy access to stakeholders that could provide the needed input/feed-back from the organization to the design team. Over time at least some of the user representatives felt discomfort meeting with their peers reporting back from the work in the ERP design team as they felt they had nothing but bad news to their peers. Since no good way of representing the organizational practice and/or the ERP protocols was established it became very difficult to find tangible ways to discuss different design issues. At rare occasions the ERP software as a prototype was used to illustrate a work process or a user interface. Hence the fact that the design works was mainly "documented" as an emerging (and rather complex) prototype had as an effect that there was no easy way to discuss design suggestions for the local protocol.

Also for the user representatives within the design team it was difficult to get a grip around something that in a systematic way could mediate between use and design. The user representatives became very frustrated when using Oracles pre-defined process diagrams because they did not reflect their own process descriptions (representations) or their way of working (practice). Thus as a design artifact the process diagrams had limited ability to mediate between practice (use) and design. Only very late in phase 2 and during phase 3 did the user representatives develop sufficient practical experience with the ERP package software to allow them a more comprehensive interpretation of the process diagrams and their associations to the ERP software and situated articulation. Thus for the user representatives the process diagrams remained a very abstract reference to the ERP software and not a way to mediate between use and design. The process diagrams, however, served as an explicit mean of cooperation; they were used as a checklist and tool for reporting progress (degree of completeness). Thus in the beginning of phase 2 the scope of the implementation was defined by the sub-set of process diagrams the functional sub-groups had to implement.

A requirements specification (as a document) is by nature a very stable document although some change requests can be added during the design process. Requirements specification is

supposed to state what is needed not how to implement it. Thus by nature it has obvious limitations as a document mediating between use and design, and it is not at all suitable to mediate between design and use. In Alfa's implementation the requirements specification is used as an explicit mean of cooperation; it serves as a check list for the ERP experts (the design team). Checking if all requirements included in the requirements specification are fulfilled can be perceived as a way to ensure that the organization get what they asked for, but it will not provide any insights about the new federal protocol or the actual design of the IT artifact. Thus at best the requirements specification provides some insight in the direction from use to design, but no insights from design to use. Yet the requirements specification is the only representation shared by the user organization and the design team that has any relation to the existing process articulation.

The emerging prototype of the software is not used in any systematic way to provide insights to the reference group, the process owners or the end users. Training material was developed after finalizing the design. It was focused on how to use the IT artifact (individual screens). Test material was also developed with a focus on the IT artifact, but almost exclusively used by members of the design team.

A manual including process diagrams representing all processes in the new federal protocol were developed, it was meant to be a manual for the user organization to understand the design of the new federal protocol. The process diagrams supplied by the software vendor were used as a basis for developing the process manual. The process diagrams in the manual remained very abstract and although word documents were used to provide more text to describe the diagrams the situated users continued to complain that they didn't understand them. Thus for people who didn't take part in the design work in the sub-groups the process diagrams had very limited ability to mediate the actual design. No one in the user organization actually used the manual, and the ERP project manager acknowledges (when I interviewed them) that in most cases the diagrams did not depict a process that could actually work in the organization.

In Alfa's organization a number of process owners had been appointed (almost identical with department managers), and before go-live the process owners were asked to sign off the new processes (the local protocol) and the design of the IT artifact. As a basis for their decision they had the process manual and the advice from the functional reference group and members of the design team. Since the process owner did not participate in the design work and no tangible way to discuss design issues had been developed during the design process then the process owners did not have any first-hand knowledge about the new local protocols. Thus the process owners had to rely on the user representatives in the functional sub-groups and the reference group to recommend/back-up signing off the solution.

The process owners ended up signing of most of the new processes although a large number of processes - many related to resource management and process management - were dismissed all together. Thus officially the process owners backed up (signed off) the design of the new local protocols but in practice they had only a weak understanding of the actual implementation (the new local protocols and the factual properties of the new IT artifact).

16.4 Mediating between use and design during federal metawork

Mediating between use and design during federal metawork focuses on reconciliation among federal stakeholders, taxonomic articulation, and design issues related to coordination mechanisms shared by two or more local protocols.

Before deciding on which ERP package software to adopt different groupings throughout Alfa's organization was engaged in creating a vision for the new federal protocol, and specifying requirements for the future software. Thus at the beginning of phase 1 before the actual ERP software was selected management and representatives from all functional areas were involved in articulating the vision for the new system and the derived work processes. Many workshops were performed within functional sub-groups and some cross functional discussions also took place. This approach made the process take place at the federal level in the sense that it involved all functional groups and top management simultaneously. The work was documented in a requirements specification, some high level process flows and business goals for each functional sub-group. The approach however had the weakness that it didn't really address cross-functional coordination or taxonomic issues. The visions and the derived requirements were developed almost exclusively within functional sub-group and they weren't challenged by the other sub-groups; no cross-functional perspective was applied. Thus from the very beginning of the ERP implementation the federal perspective had very limited attention.

During phase 2 all layers in the organization and all sub-groups in the design team needed to share insights to inform the final design of the federal protocol, and different design suggestions needed to be negotiated among situated stakeholders in order to reach closure. As illustrated in figure 16.4 this require all groupings in the user organization and design team to build shared insight of the new federal protocol. As said in earlier chapters it is especially important to focus on shared coordination mechanisms.

Within the design team some shared insights was provided; an ERP architect and the ERP project manager tried to address issues that cut across the modules of the ERP software and one configuration workshops with participation from all sub-groups was held very late in the design process. In the user organization however no natural forum took care of federal issues and no activities were performed or representations developed to help the user organization understand which federal coordination mechanisms were necessary or how to use those actually provided by the new federal software. Thus during phase 2 virtually no activities or representations mediated between use and design or between design and use at the federal level. This implies that necessary insights to reconcile different stakeholders' needs and preferences during design relayed completely on the participants in the design team.

This also implies that no reconciliation between the actual situated stakeholders was initiated during phase two. Thus no negotiation of the actual design of the federal protocol took place among situated stakeholders. This may explain why large parts of the ERP implementation were dismissed by powerful stakeholders after going live; the coordination mechanisms lacked necessary insights from use to design causing poor design, very sparse knowledge about how to use the coordination mechanisms actually provided made them fail as coordination mechanisms, and finally resolving conflicting points of views and needs was postponed until after going live.

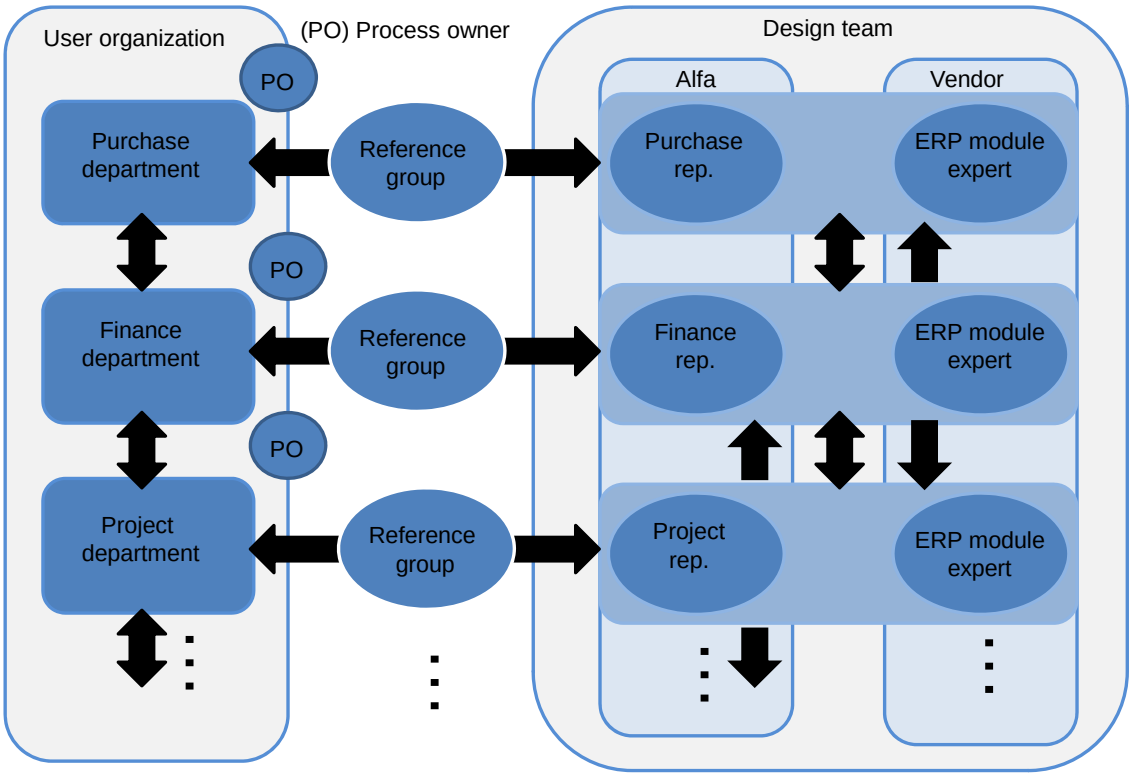


Figure 16.4: Groups involved – and need for shared insight

16.5 Mediating between representations and construction

In the ERP articulation terminology mediation between representation and construction is related to ensuring that (the model of) the process protocol build into the ERP software is reflecting the federal process protocol decided on, and ensuring that the customizations specified to the software is performed.

The nature of construction in ERP implementations are for the major parts a little different from developing tailor made information systems. Going along with the dual design objects the implementation process aims at constructing a federal process protocol and an IT artifact holding the bearing structure of the federal protocol and templates/support for articulated articulation. In relation to the IT artifact most of the construction is supposed to take place by way of configuring already existing software modules. The ERP experts are presumed to know the content of the software modules and understand how to construct different versions of the software by configuration. The ERP software is able to serve both as a design artifact (a prototype) and an emerging design object (the final IT artifact). Thus for the ERP experts there seems to be no need for mediating between representation and construction in the classical sense. The ERP experts taking part in the analysis are also the once performing the configuration (construction).

The ERP experts rely on the requirements specification and discussions with user representatives to provide insights for the construction. In many cases the communication between user representatives and ERP experts in a specific functional sub-group can rely on a shared professional background; common education or similar practical experience related to the profession in focus in the sub-group. Thus they may have a shared background within finance, purchase, project management etc.

In Alfa's implementation using the process diagrams as a model of the future IT artifact wasn't easy. First of all the process diagrams were very abstract, both user representatives and ERP experts found that at the outset the process diagrams might fit any ERP package software. Thus the diagrams didn't provide any insights how the processes was actually implemented in the software, and they didn't reflect the configuration possibilities. When discussing the diagrams with the user representatives the ERP experts found that the diagrams do not initiate the necessary discussions about details in order to make configuration decisions.

For the ERP experts using the software as a model of the future IT artifact made it possible to almost leave out the modeling step, or at least leave out mediating between representation and construction as the two artifacts slowly melt together. As described the user representatives found it difficult to fully comprehend the software as a prototype but over time, as their experience with the software increased it became easier.

The classical need to mediate between representations and construction may in the context of ERP implementations become a little twisted. Thus the (ERP package) software already exists in a semi-finished version that is expected to provide a good basis for the final IT artifact. Hence I want you (the reader) to follow me (just for the line of argumentation) and consider the consequences if the IT artifact is already "constructed" (although we know it is only a semi-finished IT artifact). If the IT artifact is already constructed and (to some extent) ready to be used then mediating between construction and use (expected use; anticipated use; use options) become an issue. How can we start to understand the options for use provided by the IT artifact?

An IT artifact can be perceived as a model of something from the use domain. If doing so what is the ERP package software a model of? As explained in chapter 7 it can be perceived as a model of an organizational process protocol; an *ERP process protocol*. Following this line of thought it is the properties of the ERP process protocol we need to understand in order to "*start understand the options for use provided by the protocol*". Actually what is needed is to understand a representation of an ERP process protocol (an IT artifact already constructed); how can such an understanding be mediated. Hence if no customization of the IT artifact was necessary then the classical problem of mediating between representation and construction could instead be understood as a problem of *mediating between representation and use*.

However as the empirical case show both configuration and technical customization seems to be necessary in order to adapt an ERP process protocol to the organizational context (achieve a desirable fit), and the nature of construction is different depending on if configuration or technical customization is applied.

In Alfa's case configuration of the IT artifact involved one primary representation namely the ERP software used as an emerging prototype. It served both as the representation of the ERP process protocol (the ERP package software as a model) and as a representation of the federal process protocol (the future software as a model).

16.6 Summing up the use of design artifacts

Applying a helicopter perspective at how representations were used as design artifacts on Alfa's implementation especially three design artifacts stand out:

- *The requirements specification* understood as the original requirements specification updated during scooping and configuration with change requests approved by the steering committee
- *Process diagrams* in different formats and versions (the original diagrams developed by the user organization as a basis for the requirement specification, the process diagrams provided by the ERP vendor and the "process manual" developed by user representatives in the design team and expected to be used by the organization)
- *The ERP software used as a prototype*

Toward the end of phase 2 two additional design artifacts were developed; *Test material* and *Training material*.

The requirements specification primarily served as an explicit mean for cooperation. The requirements specification was the foundation for the contract, and throughout phase 2 it was used as a key to resolve disputes over what factual properties should be provided by the ERP software.

A significant result of my case analysis is the fact that no formal representations of the existing situated articulation is developed. The information needed in order to configure the system was obtained by interviewing the user representatives participating in the functional group and discussing partial snapshots of the ERP prototype with the user representatives in configuration workshop.

As described above process diagrams in different formats and versions were used in Alfa's implementation. Both the user representatives and the ERP experts agree that it wasn't a success. Both parties complain about the level of details provided by the process diagrams (or maybe more correct; the level of details not provided by the diagrams), but they complain for different reasons. The user representatives find it difficult to actually obtain any knowledge about the future design of the processes from the diagrams, and the consultants on the other hand find the abstraction level too high because they need much more detailed information about the (existing/future) situated articulation in order to be able to configure the software.

As a notation process diagrams can be used to depict tasks, task sequences, task clusters and coordination mechanisms. Thus in theory process diagrams seems to have a potential to serve as a boundary object in ERP implementations. Process diagrams can be used to create abstract representations of existing situated articulation, of the ERP process protocol, of the future

federal process protocol, even comparing different versions or alternative ways to articulate the "same" process.

Theoretically process diagrams could have been able to mediate between use and design, mediate between user representatives and ERP experts, as a way to mediate a gap analysis (comparing the software as a process protocol and the existing situated articulation), and to mediate between the design of the federal protocol and situated stakeholders. However, this potential was never realized in Alfa's implementation. The empirical material doesn't give a clear answer - only a few indications allowing some speculations. Hence discharging the process diagrams developed by the organization and instead require the user organization as well as user representatives in the design team to relate to the generic process diagrams may have taken away their possibility to relate what they knew (where they are coming from) to what they saw (where the new software/protocol require them to go). Another reason may have been that the ERP experts seems to be reluctant to use the process diagrams actively; primarily because the diagrams lack a good way to relate the diagrams to the properties of the software. The failure of the process manual may have more possible explanations; the processes didn't depict the actual implementation thus was irrelevant, the user organization was not involved in the design process prior to receiving the process manual thus they had no background knowledge to interpret the process diagrams in the manual. Although no clear answer can be given why process diagrams failed as a design artifact in Alfa's case all the involved parties acknowledge that it was the case.

Hence in Alfa's implementation the ERP software seems to be the most important design artifact. During phase 2 the ERP software is being used as an emerging prototype slowly sliding into becoming the finalized design object. Again the empirical material lack the ability to give a clear explanation why this artifact is favored by the ERP experts but existing litterateur may provide some indications. In general prototypes are expected to mediate cooperation between user representatives and ERP experts in a very communicative way:

"A prototype mediates cooperation in a communicative fashion by letting users, through their exploration of the prototype, impose knowledge about their context onto designers, and by letting designers express their new insights by way of continuous change to the prototype." ((Bertelsen 2001), p. 18)

Furthermore prototypes can be used as an analytical tool (Mogensen 1994). The idea is that by introducing the prototype into a given practice it will make constrains and potentials become explicit. In an ERP implementation the prototype may be used to allow the user representatives to learn about the ERP software (ERP protocol) and the ERP experts to learn about the organizational practice. Thus for both parties it may provide an opportunity to get to understand where the organization is coming from and imagine the future (maybe a better future). This may be good reasons why the software is used as the primary design artifact.

16.7 Conclusion - answering the research question

The research question for this chapter is:

How is the ERP articulation process approached in the case organization?

The overall answer to the research question is that Alfa's implementation approach resulted in a very narrow focus on local metawork, thus all the other aspects of the articulation process was postponed until after go-live. In the text below a detailed answer addressing all the elements in the articulation process is given.

1. Specifying customizations
    a. Perform a gap analysis
    b. Decide how to bridge the gaps (customize the ERP process protocol and the ERP software or change the prescribed way to perform a work process)
2. Perform reconciliation
    a. Reconciling how (much) the ERP process protocol are allowed to constrain the design of the federal process protocol
    b. Reconciling different stakeholders' needs and point of views during the design of the federal process protocol
    c. Reconciling different stakeholders needs and point of views during the design of local protocols
    d. Reconciling federal stakeholders' needs and point of views when specifying data definitions and semantic values of data in order to obtain standardized data for management processes

16.7.1 Performing the gap analysis

The initial requirements specification wasn't expressed as a process protocol but as disconnected requirements specified by four different functional sub-groups prior to deciding on which ERP package software to implement. Based on these disconnected requirements the potential ERP vendors specified what they believed to be gaps between the ERP software and the requirements. This comparison was done prior to signing of the contract and could be perceived as the first gap analysis (although it only indirectly says something about the gap between the protocols).

After deciding on which ERP package software to implement, the cooperation in the design team was kicked off with a scoping workshop (taking about a week). The workshop can be perceived as a second gap analysis. The user representatives had to identify gaps while the ERP experts performed a walkthrough of the functionality of the ERP software; ERP experts showed the steps/screens to go through in order to cover all the requirements in the requirements specification. Hence identifying gaps wasn't based on any kind of formal representation of the existing work processes or representation of desired work processes.

At this point in time the user representatives had not yet developed much insight in the ERP process protocol. It is unclear how it was decided if a gap was identified. The user representatives participating in the workshop were shown how to use the ERP software, and based on what they saw they hat to form their own opinion whether it would work or not. A considerable amount of technical customizations was decided on at this point in time where it is clear that the user representatives have no clear understanding of either the ERP process protocol build into the ERP software or a possible future federal process protocol.

The approach used identifying gaps imply that neither the user organization nor the ERP experts explicitly recognized the significance of perceiving the ERP software as a process protocol and the implementation as an articulation process. Instead both parties focused on the functional properties of the IT artifact. The ERP experts were using the ERP software and the requirements specification as the key, while the user representatives participating in the design team had to rely more or less on their own judgment.

Summing up the approach used in Alfa's implementation I can see that it only provided an indirect analysis of the gaps between the ERP protocol and a desirable federal protocol. Only customizations in relation to the ERP software were systematically recorded and explicitly specified, thus no one was responsible for keeping record of situations where changing the organizational practice (work process) was decided on. Finally most of the decisions to customize the software were made at a point in time where the user representatives had limited understanding of the ERP software as a protocol. Actually no explicit federal or local metawork was performed.

16.7.2 Reconciliation

From the very beginning Alfa's organization had an ambivalent attitude toward how much the ERP software (the ERP protocol) should be allowed to constrain the design room. During the development of the requirements specification they focused their energy on specifying areas that they expected to be different from most organizations implementing ERP software. They also allowed technical customizations to be included in the contract prior to kicking off the collaboration with the ERP experts; before actually having first-hand knowledge about the ERP software and the configuration possibilities it provided. On the other hand the ERP project manager explained that the organization was expected to adapt to the ERP software if at all possible, and that it required extraordinary arguments to have customizations accepted during phase 2. This attitude placed some of the user participants in the design team in a difficult position. As one of the user representatives in a functional sub-group explained "*It became unpleasant to report back to the user organization because I had nothing but bad news*".

Reconciliation of different stakeholders' needs and point of views during design of the federal process protocol also seems to have been problematic. First of all the considerable amount of misfits related to federal metawork (misfit category 3 see section 5.3 and misfit category 4 see section 5.4) imply that insufficient stakeholder reconciliation was performed during metawork (or no metawork was performed). Traces of the conflicts initiated between stakeholders during phase 2 could still be found years after going live (when I performed my interviews). Also the analysis of the implementation approach and the use of design artifacts (section 16.1 and section16.2) indicate that federal metawork received very little attention during phase 2.

The misfits in category 1 (section 5.1) and category 2 (section 5.2) imply that identifying and reconciling different situated stakeholders needs and point of views during the design of local protocols was incomplete. The analysis of the implementation approach confirms that the approach used resulted in difficulties mediating between use and design, and between design and use. Although a reference group had been formed for each functional sub-group it seemed very difficult to include them in discussions. Some of the functional sub-groups had more

success than the others, but in general it was very difficult to find tangible ways to discuss design suggestions with people who didn't have detailed knowledge about the ERP prototype. When showing a snapshot of the ERP prototype to the reference group the discussions quickly became focused on how it differed from the way things were done "today". As a result the design team was more reluctant to involve situated stakeholders (the reference group). In Alfa's situation this is very problematic as local protocols are intended for multiple instances of situated articulation (not all having representatives in the design team) thus reconciliation becomes complicated. Furthermore many situated stakeholders are powerful and well articulated, not easily accepting just to do as they are told. Due to the poor design (not accommodating situated articulation) it is easy for the situated stakeholders to provide argumentation why to dismiss the design of the local protocols (part of the complex) and suspend the use of it until customizations have been performed.

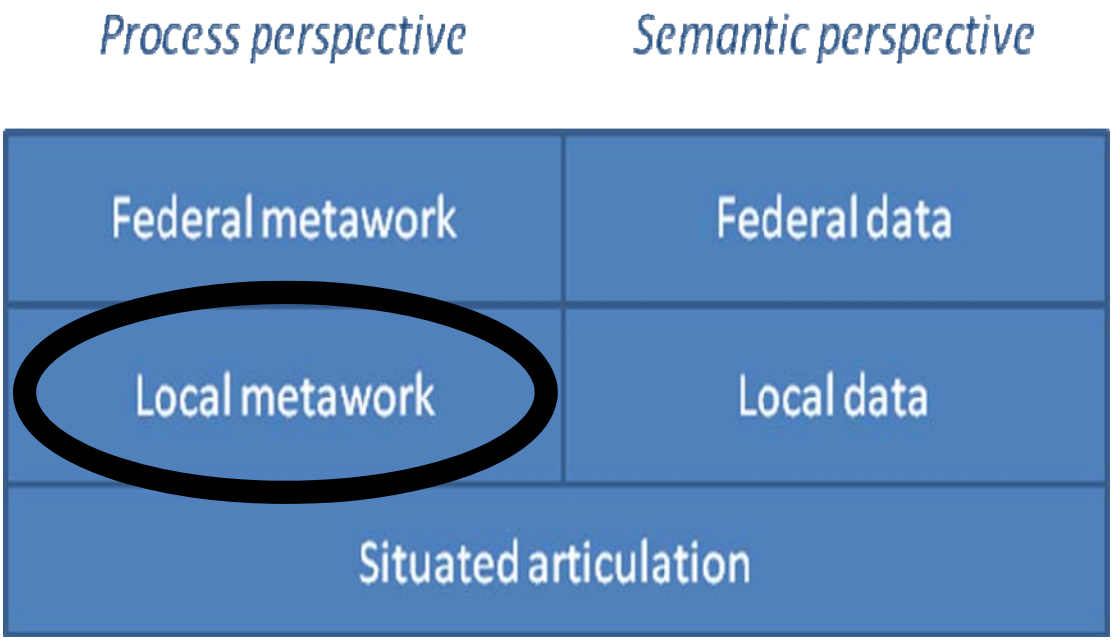


Figure 16.5: Alfa' s implementation approach resulted in a focus on local metawork

Finally, to summarize the findings the approach used in Alfa's implementation resulted in a very dominant focus on insight building within the functional sub-groups (mediating between user representatives and ERP experts). Especially the ERP experts' need for information in order to configure the ERP software seems to have been privileged.

The most important design artifact is the ERP software used as an emerging prototype that slowly transforms into the finished IT artifact. Hence the traditional separation of analysis and construction becomes blurred, probably because the semi-finished nature of the software allows the ERP expert to perform analysis and construction (configuration) almost simultaneously. Although the approach had a dominant focus on local metawork, reconciliation among local stakeholders was also limited.

Chapter 17
Reinterpreting the focus group analysis

In chapter 9 three different metaphors for (or perceptions of) ERP implementations were identified; a *standardization ware, a game,* and a *change project*. A closer investigation of the arguments provided for the *change project* metaphor actually reviled two slightly different versions of the metaphor. Thus the two sub-metaphors share an understanding of the importance of performing metawork and that metawork should be performed by "professionals". They however differ when it comes to who the professional process designers should be and how much the ERP protocol should constrain/inspire metawork. One sub-metaphor I will call *ERP change project;* ERP experts are the professional designers and the ERP artifact are the main inspiration for the process design. The other sub-metaphor I call *Business (organizational?) change project*; internal people with knowledge about how to design business processes are the professional designers and rationalization of organizational business processes are guiding the process design.

Reinterpreting the four metaphors applying an understanding of ERP implementations as an ERP articulation process, it turns out that the four metaphors have different perceptions of if (how much) explicit metawork is needed during an implementation, and how much the ERP software (ERP protocol) should constrain the design of the federal process protocol. Thus in one dimension they differ in how much metawork is expected/performed (dimension one), and in the other dimension they differ in the way they allow/expect the ERP system to constrain the design of the federal protocol and the IT artifact (dimension two). Hence using these two dimensions the four metaphors can be placed in four sections in a diagram (illustrated in figure 17.1. below).

System as designer ((I) in the diagram) is inspired by the *war metaphor*. The war metaphor is implying that the ERP software and the ERP protocol provided with the software is expected to be responsible for metawork (both federal metawork and local metawork); thus the system is the designer of the (future) federal process protocol.

ERP experts as designers ((II) in the diagram) is inspired by the *ERP change project* metaphor. This metaphor implies that explicit metawork should be performed by the ERP experts and that the ERP software in many cases will provide a good solution (best practice) that can be adopted by the implementing organization.

Stakeholder negotiation ((III) in the diagram) is inspired by the perception of an ERP implementation as a *game*. This perception implies that no actor (stakeholder) has full control of metawork; actors are changing over time, new alliances may be formed, actors may have very different agendas, the roles of the game may change over time. Thus metawork is not performed by people with special knowledge how to perform metawork or by adopting an existing process protocol (e.g. provided by the ERP system); you could say no explicit metawork is performed.

The *business experts as designers* ((IV) in the diagram) is inspired by the *business change project.* This metaphor implies that internal employees who understand how to perform

metawork are in charge of the design of a new federal protocol, and that the business perspective has more priority than the pre-defined ERP process model.

Hence there seems to be four very different ways to engage an ERP implementation. These four different perceptions of how ERP implementations can be engaged have different implications for how the roles of the different stakeholders are perceived. Below I will describe similarities and differences of the four approached in more details.

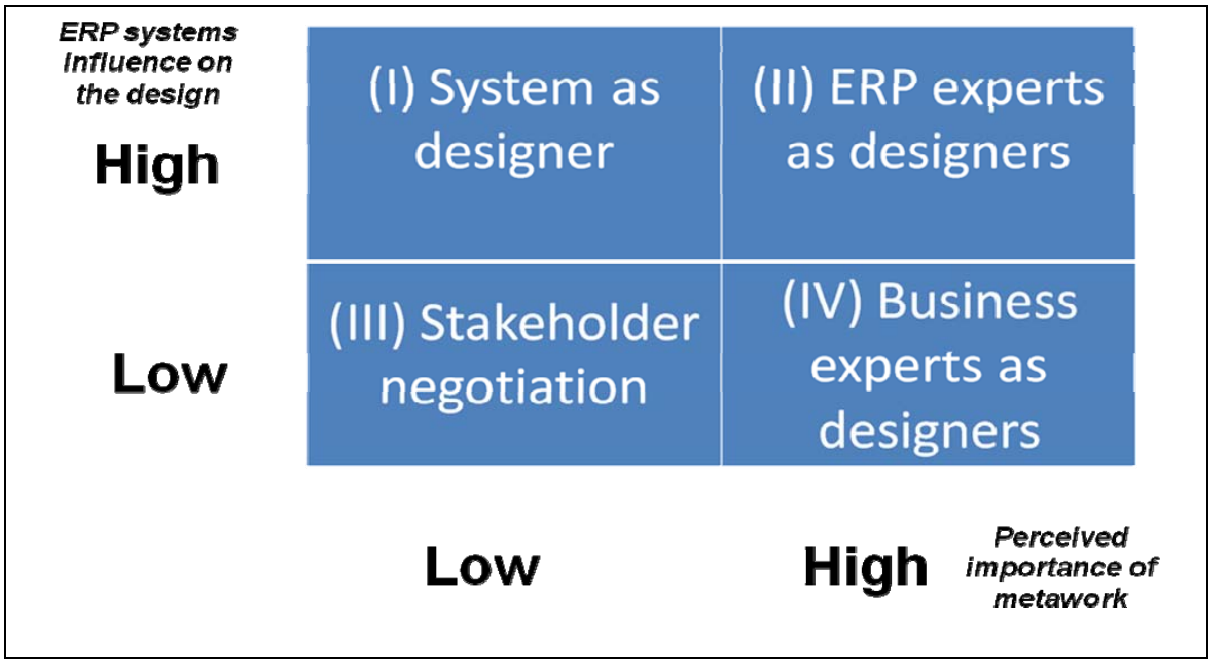


Table 17.1: The four metaphors categorized after influence and importance.

17.1 Similarities for (I) System as designer & (II) ERP experts as designers

They share an expectation that someone and/or somewhat from outside the implementing organization is given the responsibility to design the future federal process protocol. Thus the ERP protocol and/or the ERP experts are expected to be allowed a very strong position in the articulation process. For both the *system as designer* and the *ERP experts as designers* the war metaphor provides many insights in order to understand the relation between the user organization and the ERP experts, and mechanisms broad into play during the implementation. In both approaches top management is expected to decide on which ERP system to implement and then giving over the responsibility for the design of the actual solution to the ERP software/ERP experts. End users cannot be trusted deciding power when it comes to design because they are unable to escape thinking in grooves, and they are expected to resisting any kind of change. Hence the ERP experts perceive themselves as working for management making the user organization change the way they are working against their desire. Thus the ERP experts perceive the implementation as a war; a standardization war, and as the ERP experts perceive standards especially the standards

provided by ERP systems as good and desirable (standards/best practice should not be rejected by anyone) they see their own engagement in this standardization war as legitimate.

17.2 Differences for (I) System as designer & (II) ERP experts as designers

There are however also some differences between the two paradigms due to the fact that they differ in how much metawork is anticipated/performed. The *System as designer* approach more or less anticipates an "implementation" of the ERP software; no serious design work is (perceived to be) necessary. When explaining the relevance of this approach the ERP experts are emphasizing the taxonomic aspect of a federal process protocol (se section x.x for a detailed explanation of what this is about). Thus as they see it top management is not necessary concerned about coordinating distributed process, instead they have a desire to obtain standardized data, and one standard might be as good as another as long as it become a federal standard. The *ERP experts as designers* approach however perceive explicit metawork to be necessary/relevant. The rationale behind this is a focus on rationalization in the sense of doing more with the same resources or the same with less, thus when arguing for this approach the ERP experts are often using the concept of business process reengineering. The ERP experts is expected to developing a number of business cases (specifying rationalization potential) and then use them as a basis for design decisions. In some situations the business case (the design suggestion) is based on a comprehensive analysis of existing organizational work processes followed by innovative thinking about how to redesign the process, in other situations the business case is simply developed comparing the existing way of working to the perceived result of implementing a process suggested by the ERP software. However what is important when subscribing to the ERP experts as designer approach is that the ERP experts are expected to have the skills to perform metawork, and are given the responsibility developing a federal process protocol.

Despite the differences between *system as designer* and *ERP experts as designers* these two approaches have many similarities when it comes to cooperating with the user organization and choosing ways to include user representatives (or rather not including them) in the design process. These two approaches share the perception of end users as someone who cannot be trusted to make rational design decisions because they have difficulties escaping thinking in grooves, and because they have unrealistic expectations about the technological options. Thus inexperienced end users are expected to make the communication and cooperation much more difficult. Hence the ERP experts are reluctant to include end users they prefer to cooperate with someone who can mediate between the user organization and the ERP experts, someone who know the organization well and already have experience implementing information systems.

ERP experts perceive knowledge about the organizational context to be necessary in order to understand what gives rise to specific requirements. The contextual knowledge of the requirements is not meant to create sympathetic insights with the end users and their claim to have specific requirements fulfilled. On the contrary ERP experts perceive contextual knowledge as important in order to make the ERP expert able to dismiss requirements and/or provide arguments how specific requirement or process flow can be achieved in a different way using the ERP software's functionality. In chapter xx the result of the detailed analysis of the focus groups is provided, here the ERP experts' arguments are just summarized.

No matter which background the organizational representatives cooperating with the ERP experts have, they are expect to participate in interviews and workshops in order to provide organizational knowledge for the ERP experts. The ERP experts clearly prefer direct interaction (interviews and discussions) with the organizational representatives to text or formal representations. The ERP experts are not aiming at understanding existing situated articulation in details. They focus on obtaining an abstract understanding of processes and products of the processes, and they prefer to discuss future processes (to-be processes) to the existing practice. An abstract understanding of situated articulation is needed in order for the ERP experts to be able to find patterns in the organizations way of working; patterns that can be retrieved in the ERP protocol and/or recognized from other organizations the ERP experts are familiar with.

If applying the *ERP expert as designers* approach the ERP experts may also need to understand the organizational work patterns in order to (re-)design work processes that realize the full rationalization potential available in the specific organization. Thus sometimes it may be necessary for the ERP experts to visit the user organization and become very familiar with existing work processes, especially the products of the processes. Again the aim is not to become sympathetic with the end users and the existing processes; these insights are needed in order to be able to re-design rationalized work processes. Furthermore the insights are needed in order for the ERP experts to specify the gap between the new work process and the ERP protocol, and how to customize the software.

17.3 Similarities for (III) Stakeholder negotiation & (IV) business experts as designers

Stakeholder negotiation (III) and business experts as designers (IV) share the perception that the ERP software's influence on the design of the federal process protocol is weak/should be weak (or relatively weaker than in (I) and (II)). It is implied that the ERP software is meant to serve the organization, and that internal stakeholders have the deciding power when it comes to choosing between if the organization should adapt to the ERP software or the ERP software should be customized to accommodate specific organizational needs. Arguments for the "stakeholder negotiation" approach are not given in the spirit that this is a preferred or desired approach but more as calling attention to the fact that this is often the way it is experienced. Those arguing for the "business experts as designers" approach however do perceive this as the preferred way to perform ERP implementations. Thus these two perceptions of an ERP implementation give priority to organizational goals and organizational stakeholders whether it is an explicit choice or not.

17.4 Differences for (III) Stakeholder negotiation & (IV) business experts as designers

The differences between "stakeholder negotiation" and "business experts as designers" are primary related to the fact that the two approaches differ when it comes to metawork. "Stakeholder negotiation" does not explicit recognize the need for performing metawork. Thus neither the ERP software's ability to provide a process protocol or the work performed by "professional" process designers developing a process protocol is recognized. During the implementation the individual stakeholders focus on their specific work situation and how the

ERP software should be customized to accommodate their specific needs. The design of the IT artifact becomes a battlefield where different local/situated stakeholders fight to influence the design of the IT artifact as much as possible. The concept of a federal protocol is not recognized and casual articulation work rooted in local/situated preferences is performed instead of explicit metawork defining formal, explicit and standardized ways to coordinate work. The "business experts as designers" approach explicit express a desire to changing the organizations way of working by (re-)designing work processes; standardizing and segregating organizational work processes. Re-design of work processes should be performed by "professionals" with comprehensive knowledge of the organization and its (strategic) goals. In this relation "professionals" means people with the skills to perform metawork and the ability to detach them self from the idiosyncrasies of existing work processes. In general end users are not expected to have the skills for or an interest in performing metawork. At the same time ERP experts are not expected to have the necessary organizational insights or organizational sympathy to perform metawork. Thus a group of internal "professional" designers are needed.

At any point in time the molding of the process complex (the process protocol) will reflect different actors (stakeholders) negotiation power…. The four quadrants illustrate the perceived power balance between four actors involved in negotiating the design of the process protocol, the four actors are: the ERP package software, the ERP consultants, the professional designers within the organization, the stakeholders in the situated articulation.

Part VI - Conclusion

Part VI - contains the *Conclusion* in chapter 18. Here I conclude on the four learning cycles, give an answer to the overall research question as well as to the detailed research questions I have asked in the iterative learning cycles. I end the chapter by summing up my theoretical and practical contribution. Finally, I have a short section where I discuss the implication of my answer to the overall research question

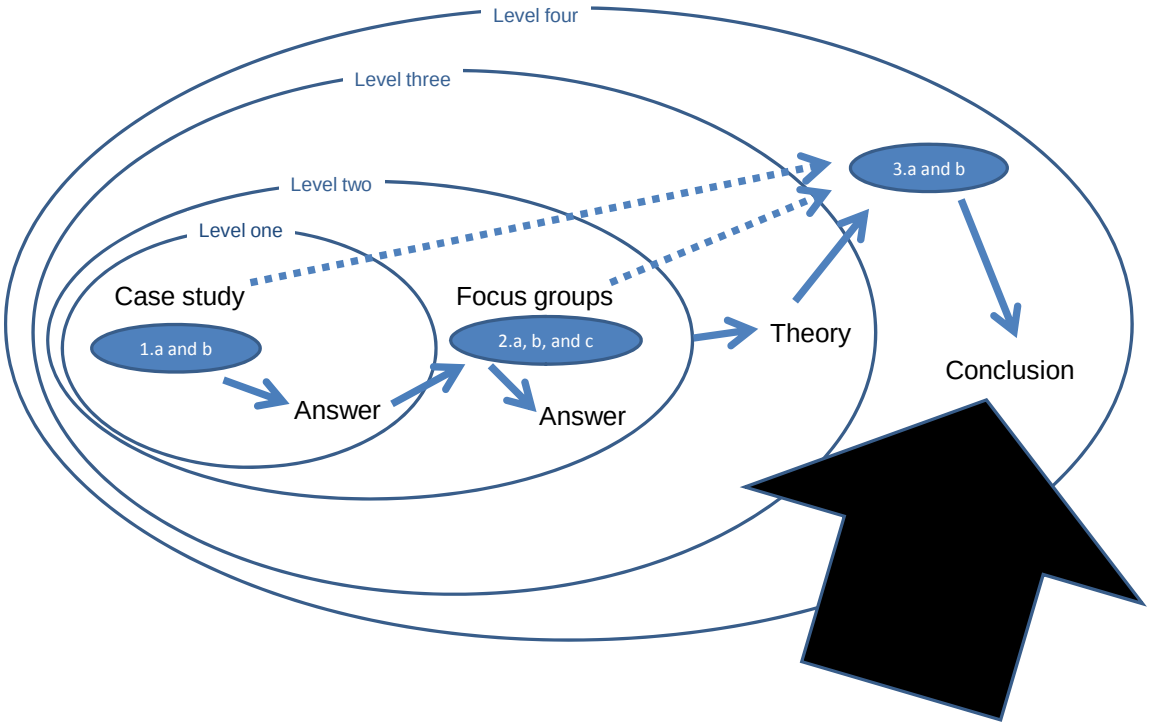


Figure VI.1: The overall structure of the Ph.D. thesis with a thick arrow indicating where part VI belongs

Chapter 18
Conclusion

In this chapter the final conclusion of my Ph.D. thesis is provided. I start out with a short answer to the overall research question. Then the conclusion for each of the detailed research questions is provided in order to give an overview of the thesis contribution. Finally I sum up the contribution; theoretically and for practitioners.

Configurable semi-finished software such as ERP package software has been widely adopted by organizations for almost twenty years. However, implementing the ERP software is very challenging and most organizations experience a serious disruption in their organization when going live, and they have difficulties recovering. Furthermore it seems as if a turbulent ride after go-live result in a long lasting disinclination toward the ERP system. These difficulties imply that the socio-technical design plays an important role in the outcome of the implementation. Organizations obviously have difficulties finding a way to make sound decisions about the mix of configuration, customization and organizational changes that will work for the organization. Existing ERP literature provide very few insights how the design team actually engage ERP implementations, and no suggestions how it could be done in a better way.

18.1 The answer to the overall research question

The aim of this thesis is to contribute to our knowledge about how organizations actually engage ERP implementations deciding on the mix of organizational change, configuration and customization to the ERP software. The overall research question is:

> How do organizations engage ERP implementations, and why does it often result in misfits?

My answer to the research question is based on an understanding of ERP implementations as an ERP articulation process (theory developed in the third learning cycle, and described in chapter 14). Using this theoretical perspective an ERP implementation is basically about arriving at a *federal process protocol* which is based on the principles of standardization and segregation in order to provide coordination and standardized data in heterogeneous and distributed work arrangements.

An ERP articulation process is performed in order to arrive at a reconciled federal process protocol. An ERP articulation process can be understood to have two different but still related perspectives: a *coordination perspective* and a *semantic perspective*.

The coordination perspective focuses on articulating operational work processes emphasizing the design and use of coordination mechanisms as a way to tie work processes together in a complex of processes; a *federal process protocol*.

The semantic perspective focuses on management processes (financial control, operational planning and decision support) emphasizing the need to provide *standardized data*. Both perspectives can be understood to have three layers: A *federal articulation layer* (focus on coordinating between local protocols and a common taxonomy for shared data), a *local articulation* layer (focus on developing local protocols for specific functions in the organization and a locally shared taxonomic) and a *situated articulation* (actors actually performing the processes covered by the protocols) layer. The different perspectives and layers are illustrated in figure 18.1.

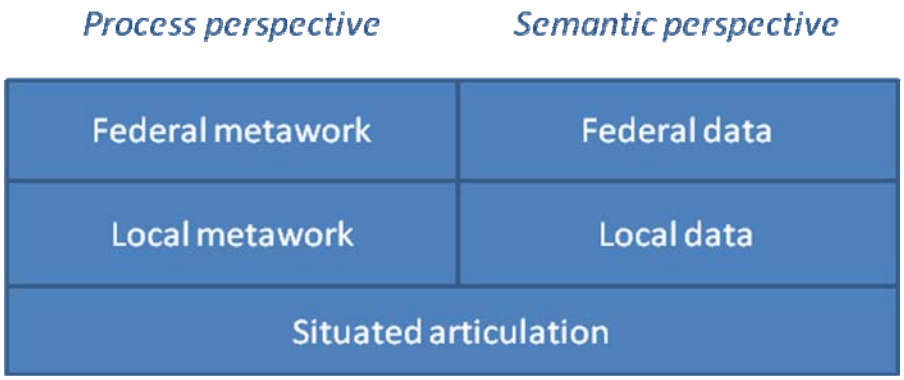


Figure 18.1: The two perspectives and three layers of my theory.

The results of this research project indicate that there may be several different ways to engage ERP implementations, four distinct approaches were identified: *System as designer*, *ERP experts as designers*, *stakeholder negotiation* and *business experts as designers*. These approaches are different in the way they consider federal and local articulation (the articulation process in design mode).

The system as designer approach is implying that the ERP process protocol provided with the ERP software (as designed by the vendor) is complete and useful for the implementing organization and therefore should be adapted by the implementing organization as their federal process protocol. Therefore, it is anticipated that there is no need to perform federal metawork or local metawork, the system is the designer.

The ERP experts as designers approach imply that federal metawork and local metawork is performed by the external ERP experts, and that the ERP software provides many different local protocols that the implementing organization can adopt. Thus it is implied that the ERP process protocol provided with the ERP package software has more priority.

The stakeholder negotiation approach implies that no stakeholder has full control over metawork, that the stakeholders may change over time and that power constellations may change over time. Thus metawork is not performed by people with special skills to perform

metawork or by adopting an existing process protocol; you could say that no explicit metawork is performed.

The business experts as designers approach implies that internal employees having the skills needed to perform process design is actually in charge of federal and local articulation work. It is also implied that business perspectives has more priority than the pre-defined ERP process protocol.

The four implementation approaches provide very different conditions for specifying *customization* and performing *reconciliation*. How the actual ERP articulation process is performed may depend on the specific local context.

ERP implementations often result in misfits because the ERP articulation is incomplete. Thus an incomplete articulation process will result in misfits caused by either a defect logical design of a federal process protocol, lack of reconciliation and/or lack of understanding how to perform situated articulation in the way it is anticipated by the federal protocol.

The four different approaches identified provide very different basic conditions for performing the ERP articulation process, but based on the research material it isn't possible to determine the likelihood of success in specific organizations. The case study in this research project however indicate that using *the system as designer* approach will be problematic as the process protocol provided by the ERP package software is clearly incomplete. The case study also indicates that using the *stakeholder negotiation* approach may be problematic as the federal perspective easily gets overlooked. Finally the case organization clearly had very powerful end-user groups (e.g. the project managers) which indicate that the *ERP experts as designers* approach would be problematic unless the powerful groups happen to find the solution provided very satisfying. The approach most likely to succeed seems to be the internal *business experts as designers* with a strong focus on reconciliation. Unfortunately the case organization seemed to lack this internal competence until after go-live.

18.2 Answer to the detailed research question on misfits

The first research question contributes to our understanding of what kind of misfits the user organization experience after go-live.

A misfit is understood as a situation in the user organization where the interplay between the factual properties of the IT-artifact and the organizational processes (according to end-users) result in:

- Users giving up using the system or introducing serious data quality problems when trying
- Inappropriate support for work (individual tasks, sequencing of tasks, duplicated tasks or dispensable tasks) performed by functional groups preventing expedite work processes
- Ineffective coordination between different user groups, and
- Difficulties using data for financial project control and decision support.

Thus the answer to the research question is that the misfits experienced in the user organization after go-live can be understood as belonging to four different categories with different properties.

The four categories of misfits identified are:

1. Misfits related to human computer interaction – individual level
   a. The first sub category contain misfits where the users are complaining about time spend entering or finding relevant information; thus it is possible to perform the task but it takes much too much time
   b. The second sub category contains misfits where the human computer dialogue makes the users give up or introduce poor data quality.
   c. The third sub category relate to obtaining a cohesive force between the ERP system, other IT systems and manual procedures.
2. Misfits related to a specific profession – functional group level
   a. The first sub-category "inadequate support for work performed by a functional group" contains design issues related to "what" work you do within a functional group and "how" you do it; the tasks performed and the sequence of tasks
   b. The second sub-category contains misfits related to internal and external data presentation; both reports and other kind of summarized data used to provide overview for people belonging to a specific functional group, and external papers meant for customers and other partners interacting with the functional group.
3. Misfits related to coordination between professions – inter group level
   This category contains misfits experienced by functional groups in relation to coordinating work between the groups. The misfits are related to work processes where some kind of coordination mechanisms are necessary to allow different actors (groups of actors) to perform their task(s) in a shared process (a series of tasks) where the tasks are split between them.
4. Misfits related to a shared taxonomy – organizational level
   This category contains misfits experienced by operational users, controllers and managers relying on data (high data quality and well understood data definitions) to perform their work.

18.3 Answer to the detailed research question on knowledge to design

The second research question contributes to our understanding of how and when knowledge to decide on the design of the new ERP system is obtained. Thus it provides an answer to the research question:

> How is knowledge to design the system obtained during the ERP implementation?

The analysis shows that knowledge is obtained in very different ways at different phases in the case organizations ERP implementation. An ERP implementation process model with five phases is applied.

In phase (1) the organization go through a process of investigating existing work processes in order to obtain an understanding of requirements they need future ERP package software to

accommodate, and to develop a vision of future (desirable) work processes. A number of business cases are formulated to support the vision. Most of the work is performed in functional groups, and as a result the requirements, the vision for future processes and the business cases are related to the functional groups. At the end of phase one cursory knowledge about candidate ERP package software (the technological options) is obtained through 2 days workshops performed by the vendor based on material (requirement specification) provided by Alfa. During phase one the existing practice is in focus and the organization is in charge of the process including all organizational levels in the process.

In phase (2) the design work is performed within a design team organized in four functional sub-groups. Only very limited work is performed across the functional groups and only very late in phase (2). The primary techniques used to facilitate shared understanding for design decisions are the requirement specification, abstract process diagrams provided by the vendor (somewhat reflecting work processes provided by the ERP package software), and the ERP package software used as a prototype. Within the functional groups the process doesn't providing much support for achieving new understanding of the existing organizational situation, or conceiving ideas about the future (how and why to work differently/change). Such knowledge has to be developed as a side effect of discussing with the external ERP expert if work processes demonstrated (using the ERP software as a prototype) can be used in the organization. The way the design work is performed in phase (3) has a didactic nature, and the knowledge development reflects it; the organizational representatives are being taught by the ERP experts. The focus is on the ERP package software, and the design process is dominated by the external ERP experts. During phase (2) neither the ERP experts nor the user representatives develop knowledge within all six knowledge areas needed for design (Kensing and Munk-Madsen 1993). The serious misfits that the case organization experienced after go live was not realized during phase two.

Phase (3) constituted the transition from design to use. In Alfa's case end user training and testing allow many end users in the organization to be acquainted with the ERP software for the first time, and the reaction wasn't positive. Especially the project managers (responsible for construction projects some amounting up to a billion Danish kroner) are not satisfied with the functionality provided by the system. They manage to get large chunks of functionality dismissed even before go-live, and after go-live they are able to convince management that serious customizations is necessary in order to make simple functionality use full. Going live the end user organization lack knowledge how to operate the system, they either give up completely or introduce data quality problems that make it difficult to make even simple functionality work for other user groups that depend on this data to be correct.

However, in phase (4) and (5), the knowledge integration capability changed dramatically. Thus when re-design is requested (re-configuration, re-definition of data or customizations) each participant covers more knowledge areas, and the communication between users and ERP experts is supported by shared experiences and a shared vocabulary. Practical experience with the system and informal cross-functional networks in the organization provided insights that allow a cooperative design approach. It is also obvious that the rhetoric used during phase two: "you need to adapt the work processes to the systems way of working" has changed, instead serious considerations on how to make the socio-technical design support the business goals takes place. E.g. if it is important to get hours spend on projects reported timely and

correct, and the human computer interface for *time sheets* (reporting hours spend on project tasks) is the reason for the difficulties, then it needs to be changed.

18.4 Answer to the detailed research question on professionals' perception

The aim of the third research question is to provide an understanding how the ERP professionals participating in ERP implementations perceive ERP implementations as a phenomenon.

> How do ERP professionals perceive ERP implementations; what are they about?

The focus group study provides three different answers in the form of three different metaphors used by the participants (described in details in section 9.7):

- ERP implementations as "*a standardization war*"
- ERP implementations as "*a game*"
- ERP implementations as "*a change project*"

The individual focus group participants is found to subscribe to one of the metaphors as their primary perception of ERP implementation, but quite interestingly all the external ERP consultants now and then agree to or use arguments belonging to the "*standardization war*" metaphor.

18.5 Answer to the detailed research question on need for knowledge to design

The aim of the fourth detailed research question is to provide an understanding of how the ERP professionals perceive the need for knowledge integration during design. The research question is:

> How do ERP professionals perceive the need for knowledge integration in order to design the ERP system?

The short answer to the question is that if knowledge integration is understood as ERP experts and organizational representatives developing shared insights in order to design the solution, then knowledge integration is in general perceived to be of limited importance. Only participants subscribing to the *"game metaphor"* emphasizes the importance.

Analytically the question was approached using two different foci; one asking about the perceived need for ERP experts to obtain knowledge about the use domain, and another asking about the perceived need for organizational representatives to obtain knowledge about the ERP package software. In very general terms knowledge about the use domain is perceived as important to the ERP experts in order to configure the system, and knowledge about the **final solution** is important for the organizational representatives in order to perform test and end-user training. However the findings are much more nuanced and some of the differences depend on whether the participant is external ERP expert or internal to a client

organization, and the different metaphors also come into play. Below a little more nuanced answer is provided.

All ERP professionals perceive it to be very important for the ERP experts to obtain knowledge about the use domain, but the arguments differ depending on if a consultant or customer perspective is applied.

The consultant perspective: knowledge is important in order for the ERP experts to argue with the client organization why the standard software is better than the existing practice or at least useful for the organization. The knowledge is not intended for becoming sympathetic with what is going on in the organization; on the contrary, it is necessary in order to be able to dismiss requirements and argue why the standard solution should be used in the organization. In general it is agreed that understanding patterns of the processes are sufficient, and if possible you should focus on to-be instead of as-is. If it becomes a matter of trying to understand how an individual user perform a specific task then you have come to fare; you need to stay on an abstract level.

The customer perspective: knowledge is important for the ERP experts to get to know the organizational practice and the organizational context in order to fully understand the unique characteristics of the organization. Thus, it is important in order to understand why the standard solution cannot be used in the organization, and in order to map organizational processes to the ERP system not the other way around.

Looking at the knowledge integration issue from the other side; asking the focus group participants how much the organizational representatives should understand about the ERP system then the analysis also discover some disagreements.

Again the disagreements are related to the internal-external perspective, and depend on the metaphors the participants subscribe to.

The participants subscribing to the *war metaphor:* knowledge about the ERP package software is of limited importance, what is important is for the organizational representatives to understand the solution. The organizational representatives are expected to specify requirements, and it is assumed to be unproblematic to provide these requirements. The ERP experts are expected to configure the system on their own. The organizational representatives are to a large extent perceived as “super users” that just need to know about the final solution a little before the rest of the organization in order to perform the final test and conduct end-user training.

The participants using the *game metaphor:* knowledge about the ERP package software is important for the organizational representatives because they need to participate in design decisions. It is acknowledged that requirements may depend on the stakeholders, and that the solution need to be negotiated in the organization, thus the organizational representatives and the ERP experts need to cooperate deciding on the solution and therefore the user representatives need knowledge about the ERP software’s capabilities.

The participants subscribing to the *change metaphor* (consultant perspective): knowledge about the ERP package is considered of limited importance to organizational representatives.

Business cases and the belonging re-engineered processes are considered the "requirements", and the ERP experts are perceived to be the once performing the design. However testing and end- user training is still perceived to be a job for organizational representatives and in order to perform these tasks they need knowledge about the solution not general knowledge about the ERP systems capabilities.

The participants subscribing to a *change metaphor* (customer perspective): The arguments are almost identical with the consultant perspective, but one important difference is that it is internal people not the ERP experts who are expected to perform the process design. The internal people are professional business process designers not user representatives, and these internal people are expected to obtain detailed knowledge about the ERP package software in order to decide on how to best design the IT artifact so that it accommodate the work (business) processes specified.

18.6 Answer to the detailed research question on design artifacts used

The aim of the fifth research question is to provide an understanding of which tools and techniques are chosen by ERP professionals for developing knowledge integration during design. The research question is:

Which design artifacts do ERP professionals use and why?

A classical division of responsibilities is anticipated. A more traditional requirements specification or business case (including a specification of to-be processes) is expected as the basis for the ERP experts' configuration and customization (design) of the ERP package software. Thus shared knowledge is not perceived to be important. But in practice shared knowledge is developed primarily using discussions with users (sometimes called a workshop). The discussions may have the character of interviewing the organizational representatives investigating requirements, or it can be in the form of user representatives giving feedback on how a process or some specific functionality (often demoed using the ERP software) suggested will work in the organization. Thus exchanging verbal information and opinions are the primary way to develop shared knowledge. The result is documented as requirements specification or instantiated as configuration decisions (often only documented actually configuration the software.

Again the research question was approached using two different foci; (1) which tools and techniques are chosen for external ERP experts in order to obtain knowledge about the use domain, and (2) which tool and techniques are chosen for organizational representatives to obtain knowledge about the ERP software.

Especially three techniques are in general perceived to be useful:

1. Visits in the user organization
2. Examples of documents used in the organization prior to the ERP implementation
3. Discussions with organizational representatives.

While the following techniques: requirements specification, as-is data model, as-is process descriptions, and use cases get very diverse feed-back, some use them and find them useful others new use them or find them unimportant.

Additional techniques suggested by the focus group participants:

- Workshop based on demo data
- Interview external stakeholders
- Management interviews
- Legislation and e.g. salary agreements
- Demo of exiting IT system(s)
- Externally provided TO-BE processes
- Rich Picture

Although the focus group participants find these additional techniques very useful they are seldom used. Interview with external stakeholders, interview with management, legislation and externally provided to-be processes are all added by participants subscribing to the *change metaphor*.

For the organizational representatives to develop knowledge about the software standard education and discussions with the consultants are perceived as the most useful techniques.

Additional techniques added by the focus group participants:

- Internal demo before and during implementations
- Workshops or prototypes on live data
- Early training and training in new processes
- Collaboration lab
- Best practice accelerators

Common for all the additional techniques suggested is that they are very costly and time consuming therefore the focus group participants find that they are very seldom used, but if used they consider them extremely effective.

Summing up the answers to this detailed research question I can see that the primary shared design artifacts used are: a traditional requirements specification, internal and external documents used in the organization, the ERP software used as an emerging prototype (sometimes accompanied by abstract descriptions of to-be processes), and especially if a *change metaphor* is applied business cases (including specifications of the re-engineered processes).

18.7 Answer to the detailed research question on articulation

The aim of this research question is to provide an understanding of how the case organization's ERP implementation is approached if using the lens of an ERP implementation understood as an ERP articulation process. The research question is:

> How is the ERP implementation approached in the case organization if considering it an ERP articulation process?

The overall answer to the research question is that Alfa's implementation approach resulted in a very narrow focus on local metawork (illustrated with a black circle in figure 18.2), thus all the other aspects of the articulation process was postponed until after go-live.

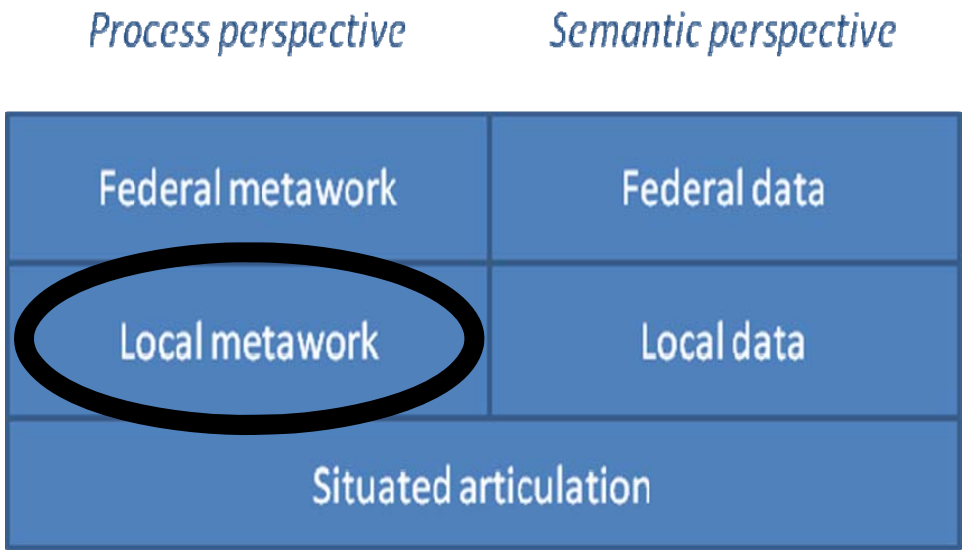


Figure 18.2: The narrow implementation focus in Alfa as illustrated by black circle

18.8 Answer to the detailed research question on perception & articulation

The seventh research question's aim is to establish an understanding of how the different perspectives on ERP implementations (the different metaphors) identified in the focus group study, relate to an understanding of ERP implementations as an articulation process. The research question is:

> How do the different perceptions of ERP implementations correspond to an understanding of ERP implementations as an ERP articulation process?

Reinterpreting the four metaphors applying an understanding of ERP implementations as an ERP articulation process, it turns out that the four metaphors have different perceptions of whether (and how much) explicit metawork is needed during an implementation, and how much the ERP software (ERP process protocol) should constrain the design of the federal process protocol. Hence there seems to be four very different ways to engage an ERP implementation. This is shown in figure 18.3.

System as designer ((I) in the diagram – figure 18.3) is inspired by the *war metaphor*. The war metaphor is implying that the ERP software and the ERP protocol provided with the software

is expected to be responsible for metawork (both federal metawork and local metawork); thus the system is the designer of the (future) federal process protocol.

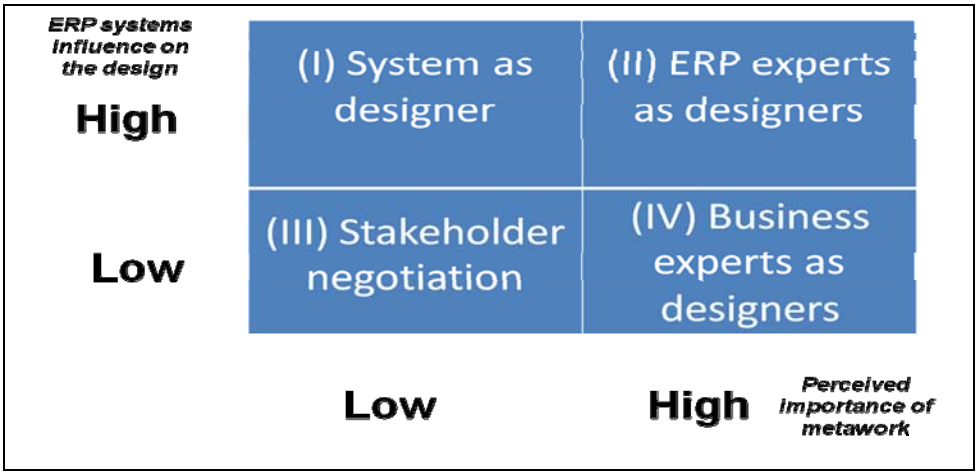


Table 18.3: The four metaphors categorized after influence and importance.

ERP experts as designers ((II) in the diagram – figure 18.3) is inspired by the *ERP change project* metaphor. This metaphor implies that explicit metawork should be performed by the ERP experts and that the ERP software in many cases will provide a good solution (best practice) that can be adopted by the implementing organization.

Stakeholder negotiation ((III) in the diagram – figure 18.3) is inspired by the perception of an ERP implementation as a *game*. This perception implies that no actor (stakeholder) has full control of metawork; actors are changing over time, new alliances may be formed, actors may have very different agendas, the roles of the game may change over time. Thus metawork is not performed by people with special knowledge how to perform metawork or by adopting an existing process protocol (e.g. provided by the ERP system); you could say no explicit metawork is performed.

The *business experts as designers* ((IV) in the diagram – figure 18.3) is inspired by the *business change project.* This metaphor implies that internal employees who understand how to perform metawork are in charge of the design of a new federal protocol, and that the business perspective has more priority than the pre-defined ERP process model.

18.9 Contributions

The research now presented contributes to ERP research in different ways.

First it contributes by adding rich insights about how organizations actually engage ERP implementation, and the ERP professionals' rationale behind different approaches.

I derive a categorization of misfits with four main types and a number of subtypes. And I specify how the knowledge needed is obtained in very different ways throughout the different phases of an ERP implementation.

As a practitioner you can use the list of misfits to reflect on how to organize especially the evaluation of design suggestions in a way that allows you to expose flaws in the design before go-live. The misfit categories may also serve to illustrate the importance of actually performing metawork (design of work processes and coordination mechanisms) if drastic customizations are made to the software. Thus customizations are likely to interfere with the predesigned coordination mechanisms. The misfit categories can also be used by ERP vendors to better understand the nature of the difficulties organizations experienced when using ERP software. Especially flexibility for modifying the user interface (support situated articulation), without jeopardizing the local and federal process protocol seems to be important.

Second it contributes a new way to conceptualize ERP implementations as socio-technical design, requiring – besides configuration and customization of the ERP software - articulation (meta) work on local and companywide level, to establish new protocols that can be supported by the ERP system. In the thesis the ERP articulation theory developed is used to defend the interpretation of the events in the case study.

My analysis revealed that ERP implementation can be perceived as ‘a standardization war’, as ‘a game, and as ‘a change project’ – 3 metaphors of ERP implementation.

As a practitioner you can use these metaphors to become more aware of your own actions as well as to become more sensible to other perceptions present in the ERP implementation. Especially the notion of a war metaphor may be important for implementing organizations to be aware of. If allowing the external ERP experts to be in charge of the design process before go-live it may result in a very turbulent ride after go-live. The implementing organization also have to consider if the external ERP experts focus on having the software up and running (within time and budget) is a valuable success criteria (Rose and Kræmmergaard 2006).

Third it contributes by discussing how different perceptions of the implementation process recognize and accommodate the necessary metawork. ERP implementation can be seen from two different angles that I called process and semantic, where process is about the coordination and mechanisms that together form a protocol, and semantic is about fulfillment of the need for standardized data. These two perspectives is then coupled with articulation and metawork into a theory of ERP articulation work. This theory includes the process and semantic perspective as well as three layers of articulation called situated, local and federal articulation.

As a practitioner you can use this theory to consider if important aspects of the articulation process are ignored when planning the implementation approach. You should consider how to address all 2 by 3 parts of the articulation process; then you are (better) prepared for a success than before!

References

Al-Mashari, M. (2006). "Enterprise Resource Planning (ERP) Implementation: a useful road map." International Journal of Management and Enterprise Development **3**(1/2): 169-180.

Al-Mashari, M., A. Al-Mudimigh, et al. (2003). "Enterprise resource planning: A taxonomy of critical factors." European Journal of Operational Research **146**: 352-364.

Allen, J. P. (2006). "Value conflicts in enterprise systems." Information Technology and People **18**(1): 33-49.

Avison, D. and A. T. Wood-Harper (1990). Multiview: An exploration in information systems development Oxford, Blackwell Scientific Publishers.

Baker, K. S. and F. Millerand (2007). Articulation Work Supporting Information Infrastructure Design. Coordination, Categorization, and Assessment in Practice. Proceedings of the 40th Hawaii Interantional Conference on Systems Sciences. Hawaii, USA.

Bancroft, N. H., H. Seip, et al. (1998). Implementing SAP R/3 - How to introduce a large system into a large organization. Greenwich, Manning Publication Co.

Barki, H. and J. Hartwick (1989). "Rethinking the concet of User Involvement." MIS Quarterly **13**(1): 55-63.

Baroudi, J., M. H. Olson, et al. (1986). "An empirical study of the impact of user involvement on system usage and information satisfaction." Communications of the ACM **29**(3): 232-238.

Baskerville, R. and A. S. Lee (1999). Distinctions among different types of generalizing ininformation systems reseach. New information technologies on organizational processes: field studies and theoretical reflections on the future of work. O. Ngwenyama, M. D. Myers and L. Introna. New York, Kluwer Academic Publisher**:** 49-65.

Bernoider and Kock (2000). Differences in Charateristics of the ERP Systems Selection Process Between Small or Medium and Large Orgnizations. 6th Americans Conference on Information Systems.

Bertelsen, O. W. (2001). "Design Artefacts: Towards a design-oriented epistemology." Scandinavian Journal of Information Systems **12**(1-2): 15-27.

Bjerkness, G. and T. Bratteteig (1995). "User Participation and Democracy: A Discussion of Scandinavian Research on System Development." Scandinavian Journal of Information Systems **7**(1): 73-98.

Bjørn-Andersen, N. and B. Hedberg (1977). "Designing Information Systems in an Organizational Perspective." Studies in the Management Sciences Perscriptive Models of Organizations **5**.

Brehm, L., A. Heinzl, et al. (2000). Tailoring ERP systems: a spectrum of choises and their implications. 34th Annual Hawaii International Conference on Systems Sciences.

Cavaye, A. L. M. (1995). "User participation in system development revisited." Information Management **29**: 311-323.

Chang, S.-I., G. G. Gable, et al. (2000). A Delphi examination of public sector ERP implementation issues. 21th Internaltional Conference on Information Systems, Brisban, Old.

Clement, A. and P. Van den Besselar (1993). A Retrospective Look at PD Projects. Communications of the ACM 36(4)

Cunningham, J., F. Dean, et al. (2001). Implementing Oracle Financial Analyzer: Delivering value-added business processes with Oracle Analytic Solutions. Harlow, England, Addison-Wesley.

Davenport, T. H. (1998). "Putting the Enterprise into the Enterprise System." Harvard Business Review.

Ehie, I. C. and M. Madsen (2005). "Identifying critical issues in enterprise resource planning (ERP) implementation." Computers in Industry **56**: 545-557.

Esteves-Sousa, J. and J. Pastor-Collado (2000). Towards the unification off critical success factors for ERP implementations. 10th Annual Business Information Technology conference, Manchester, UK.

Esteves, J. and J. Pastor (2001). "Enterprise Resource Planning Systems Research: An Annotated Bibliography." Communication of the AIS **7**(8).

Fenema, P. C. v., O. R. Koppius, et al. (2007). "Implementing packaged enterprise software in multi-site firms: intensification of organizing and learning." European Journal of Information Systems **16**(2007): 584-598.

Gerson, E. M. (2008). Reach, bracket, and the limits of rationalized coordination: Some challenges for CSCW. Resources, Co-Evolution and Artifacts. In: M. S. Ackerman, C. A. Halverson, T. Erickson and W. A. Kellogg. London, UK, Springer**:** 193-220.

Gerson, E. M. and S. L. Star (1986). "Analyzing Due Process in the Workplace." ACM Transactions on Office Information Systems **4**(3): 257-270.

Gibson, N., C. Holland, et al. (1999). Enterprise Resource Planning: A business Approach to Systems Development. 32nd Hawaii International Conference on Science Systems HICSS. Msui, Hawaii.

Gärtner, J. and I. Wagner (1996). "Mapping actors and Agendas: Political Frameworks of Systems Design and Participation." Human-Computer Interaction **11**(3).

Halkier, B. (2002). Fokusgrupper. Roskilde, Denmark, Samfundslitteratur.

Holland, C. and B. Light (2003). A Framework for Understanding Sucess and Failure in Enterprise resource Planning System Implementation. Second-Wave Enterprise Resource Planning Systems. In G. Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK: Cambridge University Press, 180-195.

Holland, C., B. Light, et al. (1999). A Critical Success Model for Enterprise Resource Planning Implementation. 7th European Conference on Information Systems, Copenhagen, Denmark, Copenhagen Business School.

Huang, J. and S. Newell (2003). "Knowledge integration processes and dynamics within the context of cross-functional projects." International Journal of Project Management **21**: 167-176.

Huang, J. C., S. Newell, et al. (2001). "The process of global knowledge integration: A case study of a multinational investment bank's Y2K program." European Journal of Information Systems **10**(3): 161.

Ives, B. and M. H. Olson (1984). "User involvement and MIS success: A review of research." Management Science **30**(5): 586-603.

Kawalek, P. and T. Wood-Harper (2002). "Finding of Thorns: User Participation in Enterprise System Implementation." Advances in Information Systems **33**(1): 13-22.

Keil, M. and A. Tiwana (2005). "Beyond Cost: The Drivers of COTS Application Value." IEEE Software **22**(3): 64-69.

Kendall, J. E. and K. E. Kendall (1993). "Metaphors and Methodologies: Living beyond the Systems Machine." MIS Quarterly **17**(2): 149-171.

Kensing, F. and J. Blomberg (1998). "Participatory Design: Issues and Concerns." Computer Supported Cooperative Work **7**: 167-185.

Kensing, F. and A. Munk-Madsen (1993). "PD: Structure in the toolbox." Communications of the ACM **36**(4): 78-84.

Khan, A. (2002). Implementing SAP with an ASAP methodology focus. San Jose, CA, Writesr Club Press.

Kien, S. S. and C. Soh (2003). An Exploratory Analysis of the Sources and Nature of Misfits in ERP Implementations. Second-Wave Enterprise Resource Planning Systems In: G. Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK, Cambridge University Press, 373-387.

Klein, H. K. and M. D. Myers (1999). "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems." MIS Quarterly, Special Issue on Intensive Research **23**(1): 67-93.

Krabbel, A. and I. Wetzel (1998). The Customization Process for Organizational Package Information Systems: a Challenge for Participatory Design. Participatory Design Conference (PDC´98), Seattle, Washington, USA.

Krabbel, A., I. Wetzel, et al. (1996). Participation of heterogeneous User Groups: Providing an Integrated Hospital Information System. Participatory Design Conference (PDC´96), Cambridge, MA.

Lee, A. S. (1991). "Integrating Positivist and Interpretive Approaches to Organizational research." Organization Science **1991**(2): 342-365.

Markus, L. (1983). "Power, Politics and MIS Implementation." Communication of the ACM **26**(6): 430-444.

Markus, L., S. Axline, et al. (2003). Learning from Adopters' Experiences with ERP: Problem Encountered and Success Achieved. Second-Wave Enterprise Resource Planning Systems. G.

Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK, Cambridge University Press**:** 23-55.

Markus, L. and M. Keil (1994). "If we build it, they will come: designing information systems that people want to use." Slown management review (summer).

Markus, M. L. and C. Tanis (2000). The Enterprise System Experience – From Adoption to Success. Framing the domains of IT management: Projecting the future through the past. R. W. Zmud and M. F. Price. Cincinatti, USA, Pinnaflex Educational Resources.

Miles, M. and A. Huberman (1994). Qualitative Data Analysis: An expanded sourcebook, Sage Publications Inc. .

Mogensen, P. (1994). Challenging Practice: An Approach to Cooperative Analysis. Aarhus, Denmark, DAIMI, Aarhus University.

Moon, Y. b. (2007). "Enterprise Resource Planning (ERP): a review of the literature." International Journal of Management and Enterprise Development **4**(3).

Morgan, D. L. (1997). Focus Groups as Qualitative Research. 2nd Edition. Newbury Park, California, Sage publications.

Mumford, E. (2003). Redesigning Human Systems. Hershey, PA, Information Science Publishing.

Nah, F. F.-H., K. M. Zuckweiler, et al. (2003). "ERP Implementation: Chief Information Officers' Perceptions of Critical Success Factors." International Journal of Human-Computer Interaction **16**(1): 5-22.

Newman, M. and F. Noble (1990). "User involvement as an interaction process: A case study." Information Systems Research **1**(1): 89-113.

Noyes, J. M., A. F. Starr, et al. (1996). "User involvement in the early stages of the development of an aircraft warning system." Behaviour & Information Technology **15**(2): 67-75.

Orlikowski, W. and J. Baroundi (1991). "Studying information Technology in Organizations: Research Approaches and Assumptions." Information Systems Research **2**(1): 1-28.

Orlikowski, W. and C. Iacono (2001). "Research Commentary: Desperately Seeking the "IT" in IT Research - A Call to Theorizing the IT Artifact. ." Information Systems Research **12**(2): 121-134.

Pan, S., S. Newell, et al. (2001). Knowledge integration as a key problem in an ERP implementation. Twenty-Second International Conference on Information Systems.

Parr, A. and S. Shang (2000). A Taxonomy of ERP Implementation Approaches. Hawaii International Conference on Science Systems HICSS. Maui, Hawaii.

Parr, A. and G. Shanks (2000). "A Model of ERP Project Implementation." Journal of Information Technology **15**(4): 289-304.

Parr, A. and G. Shanks (2003). Critical Success Factors Revisited: A Model for ERP Project Implementation. Second-Wave Enterprise Resource Planning Systems. G. Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK, Cambridge University Press.

Parr, A. and G. Shanks (2003). Critical Sucess Factors Revisited: A model for ERP. Second-Wave Enterprise Resource Planning Systems. In: G. Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK, Cambridge University Press, 196-219.

Pries-Heje, L. (2006). ERP misfits: What is it and how do they come about? 17th Australasian Conference on Information Systems, Adelaid, Australia.

Pries-Heje, L. (2008). "Time, Attitude, and User Participation: How Prior Events Determine User Attitudes in ERP Implementation." International Journal of Enterprise Information Systems **4**(3): 48-65.

Robey, D. and D. Farrow (1982). "User involvement in information system development: A conflict model and empirical test." Management Science **28**(1): 73-85.

Robey, D., J. Ross, et al. (2002). "Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change." Journal of Management Information Systems **19**(1): 17-46.

Rose, Jeremy & Pernille Kræmmergaard (2006). ERP systems and technological discourse shift: managing the implementation journey. International Journal of Accounting Information Systems 7 (2006) 217-237.

Rosemann, M. (1999). ERP-software - characteristics and consequences 7th Europan Conference on Information Systems, Copenhagen, Denmark.

Schmidt, K. and C. Simone (1996). "Coordination mechanisms: toward a conceptual foundation of CSCW systems design." Computer Supported Cooperative Work **5**(2-3): 155-200.

Schmidt, K. and C. Simone (1996). "Coordination mechanisms: towards a conceptual foundation of CSCW systems design." Journal of Collaborative Computing **5**: 155-200.

Seddon, P. B., G. Shanks, et al., Eds. (2003). Second-Wave Enterprise Resource Planning Systems. Cambridge Cambridge University Press.

Shang, S. and P. B. Seddon (2002). "Assessing and Managing the Benefits of Enterprise Systems: The Business Manager's Perspective." Information Systems Journal **12**: 271-299.

Shanks, G., A. Parr, et al. (2000). Differences in Critical Success Factors in ERP Systems Implementations in Australia and China: A Cultural Analysis 8th European Conference on Information Systems, Venna, Austria.

Shanks, G., P. B. Seddon, et al., Eds. (2003). Second-Wave Enterprise Resource Planning Systems - implementing for effectiveness. Cambridge, Cambridge University Press.

Soh, C., S. S. Kien, et al. (2000). "Cultural Fits and Misfits: Is ERP a Universal Solution?" Communications of the ACM **43**(4): 47-51.

Somers, T. M., K. Nelson, et al. (2003). "Confirmatory factor Analysis of the End-User Computing Satisfaction Instrument: replication within an ERP Domain." Decision Sciences **34**(3): 595-621.

Somers, T. M. and K. G. Nelson (2004). "A taxonomy of players and activities across the ERP project life cycle." Information & Management **41**(2004): 257-278.

Star, S. L. (1989). The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving. Distributed Artificial Intelligence, Volume II. L. Gasser and M. N. Huhns. London, UK, Pitman Publishers**:** 37-54.

Stefanou, C. J. (1999). Supply Chain Management (SCM) and Organizational Key Factors for Successful Implementation of Enterprise Resource Planning (ERP) Systems. Americas Conference on Information Systems, Milwaukee, WI.

Strauss, A. (1988). "The Articulation of project Work: An Organizational Process." The Sociological Quarterly **29**(2): 163-178.

Suchman (1993). Foreword in Participatory Design: Principles and Practices. D. Schuler and A. Namioka.

Suchman, L. (1987). Plans and situated actions: The problem of human-machine communication, Cambridge University Press.

Suchman, L. (1995). "Making work visible." Commun. ACM **38**(9): 56-ff.

Sumner, M. (1999). Critical Success Factors in Enterprise Wide Information Management Systems Projects. Americas Conference on Information Systems, Milwaukee, WI.

Sumner, M. (2003). Risk factors in Enterprise-wide ERP projects. Second-Wave Enterprise Resource Planning Systems. G. Shanks, P. B. Seddon and L. P. Willcocks. Cambridge, UK, Cambridge University Press.

Vidgen, R., T. Wood-Haper, et al. (1993). "A soft systems approach to information systems quality." Scandinavian Journal of Information Systems **5**: 97-112.

Wagner, E. L. (2002). Narrating an organisational matter of fact: Negotiating with Enterprise Resource Planning technology to achieve order within a traditional academic administration. London, UK, The London School of Economics, University of London.

Walsham, G. (1993). Interpreting Information Systems in Organizations. Chichester, John Wiley & Sons.

Walsham, G. (1995). "Interpretive case studies in IS research: Nature and method." European Journal of Information Systems **4**(2): 74.

Walsham, G. (2005). "Learning about being critical." Information Systems Journal **15**(2005): 111-117.

Weaver, R. M. (1967). A Rhetoric and Handbook. New York, NY, Holt, Rinehart, and Winston.

Zmud, R. W. and J. F. Cox (1979). "The implementation process: A change approach " MIS Quarterly **3**(2): 35-43.

Appendix A – misfits found in the interviews

This appendix contains the documentation of the detailed coding for the misfit categories.

The approach used had several iterations. First I read all the transcripts of the interviews carefully underlining (marked with yellow outliner) all parts of the text where a misfit was explained.

Then I made a new document copying all the passages underlined. In the new document a table was created; each line contained a reference ID to the original interview, the text underlined and a cell for a coding ID. Then I translated all the text in the new table to English (Table included as appendix A).

As described above more hermeneutic circles were performed before useful categories emerged covering all misfits found. I did not use the interviewees own words as labels instead I tried to understand how and why a given instance in the eyes of the interviewee qualified as a misfit.

Slowly I realized that what seemed to be important was the use situation. Thus focusing on the use situation four different use categories emerged; individual users interaction with the software in order to perform the daily work, a specific functional group's use of the software to support their work practice, the software's support of coordination between functional groups, and finally an organizational wide definition and use of data (for daily work as well as decision support). I then again looked carefully at all the misfits one by one and assigned a misfit category ID to each of them. In some of the misfit categories also sub-groups were identified. When all the categories were specified then I checked that the categories were covering all the instances of misfits identified. The result of the analysis is presented in chapter 5 in the thesis.

The four categories of misfits identified are:

1. Misfits related to human computer interaction – individual level
    a. The first sub category contain misfits where the users are complaining about time spend entering or finding relevant information; thus it is possible to perform the task but it takes much too much time
    b. The second sub category contains misfits where the human computer dialogue makes the users give up or introduce poor data quality.
    c. The third sub category relate to obtaining a cohesive force between the ERP system, other IT systems and manual procedures.
2. Misfits related to a specific profession – functional group level
    a. The first sub-category "inadequate support for work performed by a functional group" contains design issues related to "what" work you do within a functional group and "how" you do it; the tasks performed and the sequence of tasks
    b. The second sub-category contains misfits related to internal and external data presentation; both reports and other kind of summarized data used to provide overview for people belonging to a specific functional group, and external papers meant for customers and other partners interacting with the functional group.
3. Misfits related to coordination between professions – inter group level
    This category contains misfits experienced by functional groups in relation to coordinating work between the groups. The misfits are related to work processes where some kind of coordination mechanisms are necessary to allow different actors (groups of actors) to perform their task(s) in a shared process (a series of tasks) where the tasks are split between them.
4. Misfits related to a shared taxonomy – organizational level
    This category contains misfits experienced by operational users, controllers and managers relying on data (high data quality and well understood data definitions) to perform their work.

Interview ID	Misfit description	Category coding	Further comments
Interview A	"Total enterprise in construction require negative ordre. E.g. cost for cleaning the place shared by the companies working at the place". (realized before the contract)	Discovered before go-live	
	Entering and maintaining data about suppliers took too much time. (realized in the workshops)	Discovered before go-live	
	"Too many data fields and too many screens to navigate for the users. Now we are making html (self-service) for many user groups".	1a	
	(Buy goods on behalf of the customer (project)) …" The cost should be seen on the project, but not reflected in the ALFA books". (difficulty realized just before going live but wasn't resolved)	2a	
	In the purchase flow we need a project hierarchy instead of a department hierarchy. Both are needed in ALFA.	3	
Interview B	We are not buying a standard items (itemnumber) but a unique item. We have defined some high level items that can be used in the workflow authorization, but it is very difficult if the user choose the wrong item. Then you have to close	3	

	the order and start over.		
	Because we use unique service items both data and flow in the ERP system is wrong. Especially around matching.	2a	
	Error messages are confusing the users cannot understand them or what is going on.	1b	
	Two set of financial books are causing difficulties.	1c	
	The purchase order screens are difficult for the users. Difficult to understand what to enter, how to understand error messages, chose the right item/vendor. This is causing poor data quality making it very difficult to use data in purchase department and the organization for decision support. Wrong items on requests from users are a serious issue for the purchase department.	1b	
	Work around the purchase flow mean, that request for quotation is handled manually outside the system. It is primarily caused by the project manager or assistances wish to follow their requests. The request number is loosed when an order is made.	2.a	

	This work around result in the request and order to coexist. If the request is not closed when the order is opened manually the project will have a double need registered.		
			All processes we had described in the system are there at a high level, but most of them is difficult for us to work with.
			I have discussions with project people because it is possible to draw different reports that give a different impression of the financial status for the project. E.g. If we get 1/10 of an ordered delivered meaning service for 10.000 kr., then the system suggest that it is 1 (100%.) You have to manually overwrite the 1. If one forgets to change it to 1/10 then the order get marked as closed. If the project manager then draw a report of the project cost, then he cannot see 9/10 of the order is still outstanding, and the cost report is wrong. Lars made a different report providing the “correct” result. Lars think it is because the system is focused on material items (not service items) anticipating you have to deliver 1 not 1/10 of an item. With a service item we cannot know from an invoice if the “agreement note” (order) is closed (a difference between an order and a agreement note) based on the number of

			items delivered or the amount on the invoice, we have to get a massage from project management.
	Understanding the report data is difficult (they are not always taken from the place you think).	4	
Interview c	Three different needs (organizational groups) covered by the same data input. Projects (budget estimate), departments (free resources) and ALFA as a whole (management). Difficult to cover them all at the same time.	4	
	HCI – user friendliness, number of click with the mouse and such	1a	
	Supporting different user groups simultaneously (small and large projects) Small projects (project managers) have too much work and too little use of the data entered. ➔ Secretaries end up entring the data. The system we ended up with was to administratively heavy.	2a	The customizations we made means that the project managers don't have to user oracle standard functionality. (It is actually just the data base (data model) we use and reports.) Every time we have made new customizations, they have resulted in new requirements.
	There has been no natural leadership for the resource-processes. It is cutting across the organization. Timecard was needed for the financial reports, but other data has been neglected. Causing poor data quality.	4	
			I believe lack of motivation and poor user interface are

			interwoven and enforce each other.
	Timecard has been modified.	1a, 1b	
Interview D			Asked if the system is best at reporting up toward management or supporting the project managers Mikael answer: It is not good in any case. It has to do with both the data in the system and the facilities (reports) to get data out again.
	Project managers use the system very little and even if we come out and help them again and again, they give up. The user interface requires a super user. If you are just using it from time to time you don't know how to use it and the system is not helping you.	1b	
	Timecard in the new version is ok, but if you as a project manager want an overview of the project it is impossible. Is it because data is missing or because of the HCI? Especially the projects converted into ONE are problematic because you have to look more than one place for data. But also just getting an overview. I don't know if it is the ability or will that are the problem but they give up. What they need is one place where	2.b (4)	Time is very important in the projects therefore their patience is limited. If you need to spend 30 minutes finding a solution you give up. (ALFA – NN relation is mentioned as a difficulty because you have to look two places and add the numbers. If it is a converted project you have to look three places.) If it is a fixed price project or a project for another customer then NN then everything is working.

	they can enter the project number and the date, and then a financial statement comes out. Do you have the data? Is it possible to provide the functionality? The data is there but you would have to buy something that provides the functionality. I think the project managers are right being reluctant to make the statements on their own. There are so many difficulties you have to be aware of to get everything.		
	If people in the organization use the system correct it is actually possible for us to generate a correct/fair picture of the financial situation. But here the user friendliness kicks in e.g. around purchase. 50-60% of the costs on a project is purchase and most of the users don't know how to do it correct. They did it before so here it is a large step back. What is difficult?	1.b	

	It is simply all the different steps you have to go through when you want to buy something. You have to know a lot of information in advance, and already after the first 2 or 3 steps you are lost. It is especially the error messages that cause you to give up. Sometimes you have to know that you need to push a specific button, you don't get any help from the system.		
	If we have do provide a detailed specification of the invoice to the customer then we often lack the necessary information. I thing data is in the database but not specified on the invoice.	2.b	(Time, quality and money are linked – maybe we did save some money to begin with, but I thing we pay them later in the project (ERP project)). I think it is the costs that are not visible in the ERP project that are greatest. We spend much more time than necessary on the project. We cannot perform as we did before.
			Misfits after going live e.g.: I immediately discovered that no "." was used to dived the numbers.
	Notification flow required people to be present all the time. (the VP trust that we made the budget correct anyway thus we could be allowed to accept) Sometimes I call the financial department	3	

	and make them accept. I understand that an approval is necessary if I change the budget up or down, but moving money from task A to B, not notification flow should be necessary. The notification flow for purchase however works fine.		
	Sometimes I have to enter the same information two or three times. I think ONE is like a picture that is a little unclear, I can see what it is, but it is not clear.	4	
	We have been in constant contact with our ONE solution center, and some things has been changed for the better, but to solve the more serious issues the system need a re-setup. E.g. concepts are not defined clear enough. In the heart of the system it is defined wrong or we have chosen the setup careless. As a result a committed post does not mean the same different places in the system. I believe it is because the system is setup by different people. But I don't know enough about the system to be sure where the problem origin. This is a general problem as user, you always miss the last 10% to fully understand.	4	

			In general you have to be very carefull a small error when entering data on the project and the result is way off. E.g. if no turnaround is generated on a task a small warning light should be turned on.
	Allocation of people on a task is causing problems. You cannot enter time on a time card before the notification flow for resource allocation has been completed. (all you want to do is give the department head some turnaround)	3	
	We also lack proper reports for the external customers. A report has been developed, but I don't give it 6 month, it has to be changed. The report is specified by our VP John Fransen. It is possible to make manual invoices and they are not included in the report.	2.b	
	Cost budgets are another thing we miss,	2.a	
			Now we are using more time maintaining information in the system than on looking forward. It is a change in our work forced by the system.
	We need many different reports.	2.b	
	The ERP systems ability to support presentation of data is to poor. (Cognos is used instead) Especially when we need data from the departments.	2.b	
		2.b	Almost all reports are exported to EXCEL.

	We lack a report that can show turnaround including the converted data for a specific project, package and task including the committed plus estimate to completion. Furthermore a comparison to the budget. I make one every month, but I have to export data to EXCEL and ad data manually.	4	
Interview E	Allocating many people on a project has been time consuming and not working in the organisation. We have now made a customization allowing "mass allocations".	1.a (3)	
			The *project manager* had no intention to go in and approve anything. And some of them had not had training. It was well thought trough and set up, but it was bypassed by the project managers. We had to give training before the system was finished and tested. Another problem was that the report functionality was cut out in the beginning of the project. It is what makes It valuable for the users, it is their motivation! For me it is natural that it takes time to enter data in the system if you want control. Someone has to be very focused on who is working on the project the next day.

			We made a process flow (notification flow) where the project manager could do some things and the department manager other things. It is based on roles and there are more roles including managers on different levels. If people don't do their part of the flow, then it all stops. It was well thought trough, but the organization wasn't ready to become so structured (following rules).
	We had a problem with the change order logic. Scope change in a project. It is a very important and difficult issue, when we discuss with the customer if he got what he was entitled to. It is now a part of our controller manual, with 20 different areas that we find important. But there are also some trues to the difficulty using the system. I think it applies for all larger ERP systems. Even if you know the system very well it takes time. Even with smaller systems like Navision that I have worked with.	2.a	In my point of view it is absurd we have spend so mush time on perfecting it, when we cannot even get the basic cash-flow out of the system. It is very basic and your motivation for entering the budget on a monthly basis. If I cannot get my data out!!!! It shall be possible to get data out to EXCEL or another tool we can work with. Our problem is, that we named the ERP project ONE to illustrate, that the system should be the ONLY ONE system in ALFA. That leaves to many open rooms. We have had mega long lists of functionality we needed/wished for and no one could keep track of them and priorities. Could dynamic in your data on the projects be a reason for project managers to be reluctant to update them? It is properly 50% of the explanation, but the other 50% is that they don't have

			any understanding of what financial control requires! People (project managers) believe financial control comes automatically from pushing a button. Many of them are not trained to work with financial control. The organization is not ready.
	The difficulties getting the right data out of the system and be able to present them in a useful way are frustrating. I will always have to use EXCEL when making reports to management (project management).	2b	
	Data is there but you cannot work with them? Yes, it is our experience that even very small changes take more than 6 months. And when it comes to data quality then we are not much better off than with EXCEL. You should not have discussions about what costs the system use when making a profit-lost statement, where to get the data in the system. It is not difficult telling what data we want, but it is difficult finding the right place in the system to get them.	4	
	Another thing taking very long to get is a simple “dækningsbidrag”	4	

	for the project. This has caused anxiety in the finance department and a very negative attitude.		
	Cognos are now used for management information. But data quality is still a problem.	4 + 2b	In the organization it has become legal not to use the system. The story goes that the system is difficult and you should have pity on me. People were able to get out of training and using the system because the common understanding (history) is that you should have sympathy and pity with the users (project managers). (Lene: what you say is; that no one higher in the organization demanded the data before?) The demand for data is coming now – very strong. It is very classic, if the economy is good no one cares, but if things are getting more difficult than you would like to know why and what are your options. Fixed price projects and external projects are now changing the organization.
	I don't understand why we have decided not to have the project dimension used in GL. If we had done it, then we would have had the profit-lose statement directly. It is the biggest mistake we have made designing the system! I cannot make the system balance 100%! It can e.g. be materials that we sell (vidre salg) –	2.a	

	why do I have to look at PL data instead of GL data? Steen and XX explain that they see the matter from two different perspectives. Steen sees it from the organization as a whole (GL) and XX from the projects.		
Interview E	We only use the ERP system to get realized costs. We have almost the same situation as four years ago.	4	
	Resource allocation e.g. which is very interesting for us because it is a very large part of the cost and revenue. When you are allocating resources in the system you need to use 4-5 different screens for each person. If you have 2-400 people on a project and need to revise the estimate at least once a month, than you need a full time employee alone for that task.	1.a	Finally we now have had a customization implemented, it took more than 6 month to develop it and it is still problematic. Lots of problems resulting from the customization, blocking/locking others, I have difficulties understanding, that no one before us had this problems, but that is what I am told when asking and wondering. I have been close to contacting other companies myself, but then realized that it is the wrong way to go, I simply have to believe that I get a correct answer when I contact the solution center. But often I feel that I get a "standard" answer, and they hope that I then will be quiet for a while.
	We also went live without basic reports. Which meant we were forced to make some EXCEL solutions? People complaint about the reports in ONE. They find them impossible to understand.	2.b	There is no doubt that the hesitation or resistance toward using the system is partly caused by lack of training. Knowledge has not been defused; it is concentrated on a limited number of people.

			(page 2) Early training using a system with many errors or a demo system not providing the right functionality left a lasting impression of a useless system although it later became much better.
	Realized data in the system is ok, but the estimates; the budget data. Changes happens fast; new people are added to the projects, new orders ect.. Maintaining data is too difficult and time consuming. It is impossible to make cash-flow analysis. Therefore EXCEL are chosen. That allow people to work with templates, and quick updates are possible.	2a	What is needed is a screen with people in one dimension and allocation in another dimension. It has to be quick. That was also the intention with the new customizations, but the result is not good. It is no fun entering and maintaining data all day long. People are well educated, they want a challenging job, and they want to use the skills they are trained for. We have a very qualified group of controllers; they don't want to spend all their time entering and updating data!
	Stakeholder interests are pointing in different directions: E.g. We have some projects using a foreign currency. Some of the reports we get can only operate with dkr. If It should be possible to get them (invoices and statements) in any currency than someone else have to update the exchange rate instead of getting the amount automatically as they do today. It would have a negative effect, making work more complicated than it is now. These	2a	There is no doubt that finance has had a top priority in the implementation, controlling has always been secondary. But now it seems as it is time for others to get something. Controlling went from 2 persons in 2002 till 20 in 2007. New focus caused by strategy (more external projects and fixed price projects), financial difficulties and fluctuations in demand for resources. Especially after we had two large projects hanging dangerously over the cliff. That resulted in a demand for more and closer financial control. NN buy financial control in addition to project

	kinds of truckles (fights) are going on all the time, who should have their need prioritized?		management form us. During the ERP project Michael Chefer was the manager of the "project-center". At that time project assistances and controllers were not really a profession. He is an engineer and I am an economist, and back then they only looked a little at the numbers now we are much deeper in the material, there is no comparison. In 2005 controlling was written into our quality system. It is a requirement that any project 20+ millions have to have a controller allocated. The controller will influence the project manager and continue to follow up until he has the information he need. Thus it is the controller not the project managers who carry through change. A few project managers may have been engaged/convinced but most of them expect the controller to take care of it. Are the data of no importance to the project manager in his daily work, does he have an incentive to participate? Absolutely, he has to provide financial statements to the board including cash-flow for his project. From this quarter a profit-loose statement are also required. It will become a apis-goal; a bonus goal for the project managers. They need to reach a specific DB for the projects. This will make those hesitating; those thinking controlling has been taken to

			fare, wake up. Management want make profit-loose statements for all projects now to be able to compare projects and understand which we lose money on. Therefore resource management becomes very important, and the ability to estimate the resources allocated to a project (skill, cost ect.) This is another area where Brian and I work together. We sit together and go through the system exploring which difficulties we run into; completely hands on.
	We use many EXCEL and ACCES tools. The disadvantage is that we spend much time moving data around. We use 70-80 of our time generating data; manually pulling data from all directions. This of cause gives a great risk that something is missed.	1.c	We have a weekly report where we draw data from ONE and import them into EXCEL, making linear predictions with warnings etc. using macros. It works fine.
			We made reports and sat up warning lamps, but until now the demand for the information has been limited and when no one react they become a joke, and therefore we took them off again. Management did not follow-up.
	Regarding resources we are now getting to a point where we can enter data about allocations, but we cannot get a report out showing the resource needs over time e.g.	2b	

	budget and realized data. But I have an EXCEL report based on macros that will do it now.		
	To begin with we had to enter the budget and periodize it monthly. But if you have 400 tasks for each period and a project last 18 month, then you have 18X400 fields that has to be updated every month.	2a	What do you do in your EXCEL solution? We divide it equally monthly or we use a pattern (percentages) we have experienced work well distributing the allocation automatically over the project lifecycle. It is a budget covering a very long time period, but our historical data give a good indication. When it comes to projects they have to major elements; a resource part and a purchase part – time spend by our own people and external purchase. We do resource budgeting and purchase budgeting in two different processes. I have not understood the difference between self-service and the application. It is not logic what is covered by what, and I cannot get an explanation I understand. No doubt the users reluctant to use the system are caused by lack of knowledge about its possibilities and limitations.
	We had a situation where a project with 100 persons was budgeted correct and then a department manager was able to add 3 men 100% and all our work was wasted. They only needed to contribute a	3	Brian has been a controller previously (now in the ONE solution center) and we try to help each others through. He helped us get solutions in EXCEL, not in ONE unfortunately, but it works fine.

	few percentages in a few weeks, but it was much easier for the manager to enter them 100%, and he had to add them to allow them to report time on the timecard. Therefore we had to require that allocations are locked, and if changes have to be made they have to go through the controller/project. This caused problems when new data entrance screens were made recently because it locked for a lot of other things. People could not report time on projects they were allocated to and so on. Then you think good lord this is challenging. As an ordinary user of the system you are shocked. Ideally the users ought to be the designers but it is so complicated, and at the same time there are so many conflicting needs.		He is the security line for us in controlling and the people in finance. The organization would have been split into atoms. It is a lot to require form one person. He is also giving short seminars on Mondays where he goes through the system and explains it to us. It is an unofficial arrangement, no of us think it is ideal, but nothing is coming from the official management channels. It is something we do because some of us know each other on a private basis.
			Would you have been able to define these requirements during the implementation? No; it is impossible to predict, much many changes has been introduced form the top management. Everything is more structured then two years ago. Also new rules for companies registered on the American stock exchange. Because we buy parts for NN it also hits us. You have to have more strict procedures; templates and you have to

			document what you are doing with signatures. Much more control than before.
	In the old system you could get an overview over the financial status of a project. Now we have to look 4-5 different places. It is also a problem for project managers on small projects. ONE has a report, but does it show the correct result? There could be an issue about currency, therefore I have to remember to look there ect…..	1.c	Finance is not their primary area; therefore they need it to be simple. When they try to get their needs recognized by the solution center they get rejected, they are told it is not possible in an ERP system and here the conflict goes again.
			Are your requirements still developing fast? No not as fast as the last 2-3 years. Now we have a more structured approach and we try to do the same across projects. The procedures have been specified and inscribed 117 different places. At the same time I can see, that we are in front of other large companies when it comes to controlling. We have reached a point where it gives meaning, where we are able to specify what we need. Now I have Brian to play ball up against. My main focus this year will be to get ONE included as much as possible in the controller manual. It is about time put the past behind us and move forward focusing on making it work.
			How close are you to understand the abilities and constraints of the system? Rather fare from understanding. I understand

			how to operate the system from a user perspective, and when you look at it it seems fine. But I think you need an even deeper understanding, you need to work very close with it. My controllers are told to ask Brian if they have questions, and then they get advice which reports to use. They still lack confidence figuring out things on their own.
			Before I became manager of the controlling department no one had time to think about procedures. As a controller you are allocated 100% to a project. Maybe if someone as me could have participated in the implementation it could have been used as a leverage to standardize some things. Maybe not at the level where we are today, but we could have started the work.
	We have issues with currencies and vat.	2.a	
	We have decided to place much of our procedure outside ONE because we couldn't make it work. The customers and management were shouting for reports and the system could not provide them. We managed to find a way around the system, but today a critical inspection is needed. It is my focus in 2006 to make more use of the system, get more value out of the investment.	2.b	It is my impression that we have data in the system. I need a more visible structure making it possible to see where data comes from and how it is aggregated.
Interview F			After going live we got some millions and formed a project

			to improve the system. We went back to the users and looked at their processes. All wishes were noted. And afterward they were grouped and prioritized. We knew what was important for the users, but we had to give the very difficult issues a lower priority. We had a project owner from the top management group (direction).
	One of the mini projects was about an easier way to start (oprette) a project. We made a new web-interface using an API. It was only a matter of interface issues. It is a simple interface where looked up values are carried over.	1.a	We had Oracles system architect to participate in the meetings with the users. He was pulling out the last bit of hair. “Can’t they understand it is a good way to do it???” No they can´t it is not logical for them! Now we are giving a round of mandatory education where they users need a signed exception if they don’t want to participate. (The Oracle solution manager has a KPI that the satisfaction with the system has to improve at least 10% a months after the training).
	We had issues with notification flow e.g. invoice. Some people refuse to lookup the notifications in ONE. They perceive it as an e-mail and they only want to read e-mail in one system! Thus we had to make a change tricking an e-mail when a notification was created.	1.c	If we had made real BPR we would have analyzed how to get all the “invoices” (rekvisitioner) into the system upfront in the process. And then considered change management as a part of the design. But we didn’t. We just said: we need a system that can automate this if people put the data into the system. That is why we still have purchase that never makes it into the system. See here I have an example; every second week I get a basked of fruit. I never entered the request myself, it is something

			I ordered 2 years ago and I still receive a manual invoice in a green envelope. It is a matter of where the implementation projects responsibility end and the departments take over. In my opinion it is the responsibility of the finance department. We have not checked that we have process owners. During the implementation we had a responsible for all processes, and at a point in time we also had a process-integration responsible. Who should look at all the processes and insure they fit together. There are a lot of overlaps or interfaces between processes. We used Oracles tool "Tuter". (more data about the process integrator page 8)
	Time-card issues could not be resolved. Projects have numbers and names. When a manager has to approve a time-card only the number can be shown. An when a manager approve 25 employees projects all they can see it 50+ different numbers. It is impossible to remember. If they had seen the names, they would have known if it is correct or not.	2.a	We wondered how difficult it could be adding the name to the notification, but it is impossible. It has been all the way to Oracle in USA. It is impossible in this version of the system. When we upgrade the system it is possible. Tuff luck it is an issue mentioned by almost everybody as a major problem. It is rely sad not to be able to provide a solution. It is typically a secretary who has the problem because they approve on behalf of the managers.
Interview G	A large customization around flex-time. (as part of the contract)	Resolved before go-live	
	The system is only used as data storage.	2.a	
	The "kontoplan" is too large and complicated, and the customization is	4.	

	causing problems with the original functionality in the system.		
	The support for the project managers is not good in the system as it is today.	2.a	
	Cost control for the projects are difficult because of the organization operates as a matrix organization, and historically the departments (not the projects) have been the focus of cost/revenue control. (more details on page 5)	2.a	Because the organization historically has operated according to time and material (not fixed price)what is interesting is how many hours each department sell to the projects, not the cost of the project itself. (Bemærk at den historiske kontekst her spiller en meget stor rolle for hvad der opfattes som rigtigt og forkert. På side 5-7 er konflikten og konteksten beskrevet, der er mulighed for en nærmere analyse)
	We (the IT department) has been forced to make reports where they are able to change the data them self.	4	Bo thinks it is wrong. "You should get the data from the system, and accept they are positive or negative according to the reality, no be able to adjust data as you please." (Page 5)
	We now have a situation in the project module where it was decided during the implementation to generate a separate cost line for each hour. It gives enormous amounts of data in the database and the project managers would prefer to see it as a column instead. But it will require a lot of work to change this in the system now. Furthermore	2.a	The time pressure made further argumentation impossible and as consultants we did what we were asked.

	customizations have been made based on this decision.		
	The users have not been able to get the data they use to look at. Therefore a large number of reports have been developed.	2.b	
	We have experienced that many customizations have ruined data in the system because they haven't been tested sufficiently. E.g. a "overtids tilpasning" extra time customization, it has changed data making it impossible to get a financial over view of the projects and we got wrong data in the main book. Therefore we have had to make data fixes and changing code.	4	It is a customization that has been developed by Oracle, it was not tested well enough and that has been very expensive for ALFA. First Oracle tested it and then ALFA tested it according to internal test scripts.
	Flextime is another large customization. (part of the implementation)	Resolved before go-live	
	A lot of new forms to enter extra data on projects e.g. KPI information. They have a form with 10 new "faneblade" allowing the project manager to enter many, many data about the project that can be printed on a report to the customer (PSR report). The data is needed because ALFA is very special in the sense they are the project manager	2.a	The possibility of entering the data has been ther since go live but the report has just been developed (two years after going live). The complexity of project management in ALFA was described in the requirements specification, but it took a while before we realized the implications or relay understood the description. Here was a major difference in what we originally believed we were to deliver and what ALFA expected.

	for their customer. Therefore they have to look of the project form a customer perspective although they sell hours/employees to the project.		It was rely difficult to understand that they had such a close cooperation with the customer. This resulted in the solution with two set of books and many reports to provide an overview. A copy of the financial book and the functionality around it was taken and made simpler to meet ALFA´s needs. It is interesting that many of the customizations we made for ALFA is now becoming part of the standard. It seems that there were others in USA e.g. who also had the same needs as ALFA.
	First of all finance consider it a financial system and are very offended it is not possible to see all financial data at a snap with the fingers. And the project managers are offended it is not possible to look at the project from the customer perspective.	4	
	Oracle functionality is meant for controlling your own projects.	2.a	
	Scaling the system to support complex as well as simple versions of the same process.	2.a	I bring the issue up because Pernille mentioned it. Bo: I think it is a very common issue. You make the system to work well in a specific situation. Oracle has some ways to deal with the problem,

			and some of them we use, but it is not possible for ALFA to take advantage of them all because they have two set of books. If you were an ordinary project organization some functionality would be provided. Processes in finance and purchase are the same every time. Another customization has been made to provide simpler versions for ALFA. The two books means that all projects has to be entered twice (the ordinary financial book and the fictive book) and linked to each others, and then use the specially developed reports showing the ALFA view.
	Resource allocation is another example of a process that has difficulties scaling. You need to go through way to many steps before you have a correct allocation. Therefore a customization was needed to make it more simple.	2.a	In this case the solution provided by Oracle is not useful for ALFA (and maybe not for anyone). We went to a seminar with Oracle in US and they admitted it is a very difficult (clumsy) process. How did you in general decide whether to keep the Oracle process or make a customization? It depend if it were during the configuration or after go live. After go live all users were interviewed and out of the process came a long list of change requests. A new screen for resource allocations was one of them. During the configuration it was all about

			how much does it cost and do we have the time? We just cut of things. A year later more money came on the table and it became possible to get some of the things you missed which had been cut off because it couldn't be afforded during the implementation. Did you change the perspective on the use situation after going live? No…. I think ALFA was very focused on making things easy for the users. Did you see any strategic decisions requiring the system (and the users) to operate in a specific way? No I think ALFA focused on the users and the requirements cam bottom up. But now I see a change, management requires from the projects that they have to look at the cost of the projects. This is also defused to our department, that we need to ensure the functionality is there, the data they need, and that the functionality is used. It has been prepared form the beginning, but hasn't been used until now. Resource allocation has been hanging a little in the air. Sometimes it has had the support form management and other times it hasn't. It has never had the support long enough to actually be taken into use.

Appendix B section 1

Appendix B has four parts:

Appendix B section 1: A small part of the transcripts from focus group 1
Appendix B section 2: The timetable and questions for the second focus group
Appendix B section 3: Diagrams used in the focus groups to evaluate the techniques
Appendix B section 4: Detailed schema for the three metaphors and roles of different stakeholders

Appendix B section 1

Below a section of the transcribed text from the first focus group. It is included in this appendix to allow the reader to understand the approach used to derive the conclusions from my empirical material. Although it is in Danish I hope it is useful as an illustration of how I have been working. In the text below you can see the ID that refers to in the thesis in chapter 9 e.g. in the thesis' section 9.1.1 I make a reference to (Focus group 1 section 50), this means that I used section 50 from the text below. Whenever using a specific citation I had to translate it.

…….(Some discussion has already taken place in the first focus group, we jump in at the beginning of the discussion belonging to the analysis used in section 9.1.1 in the thesis.

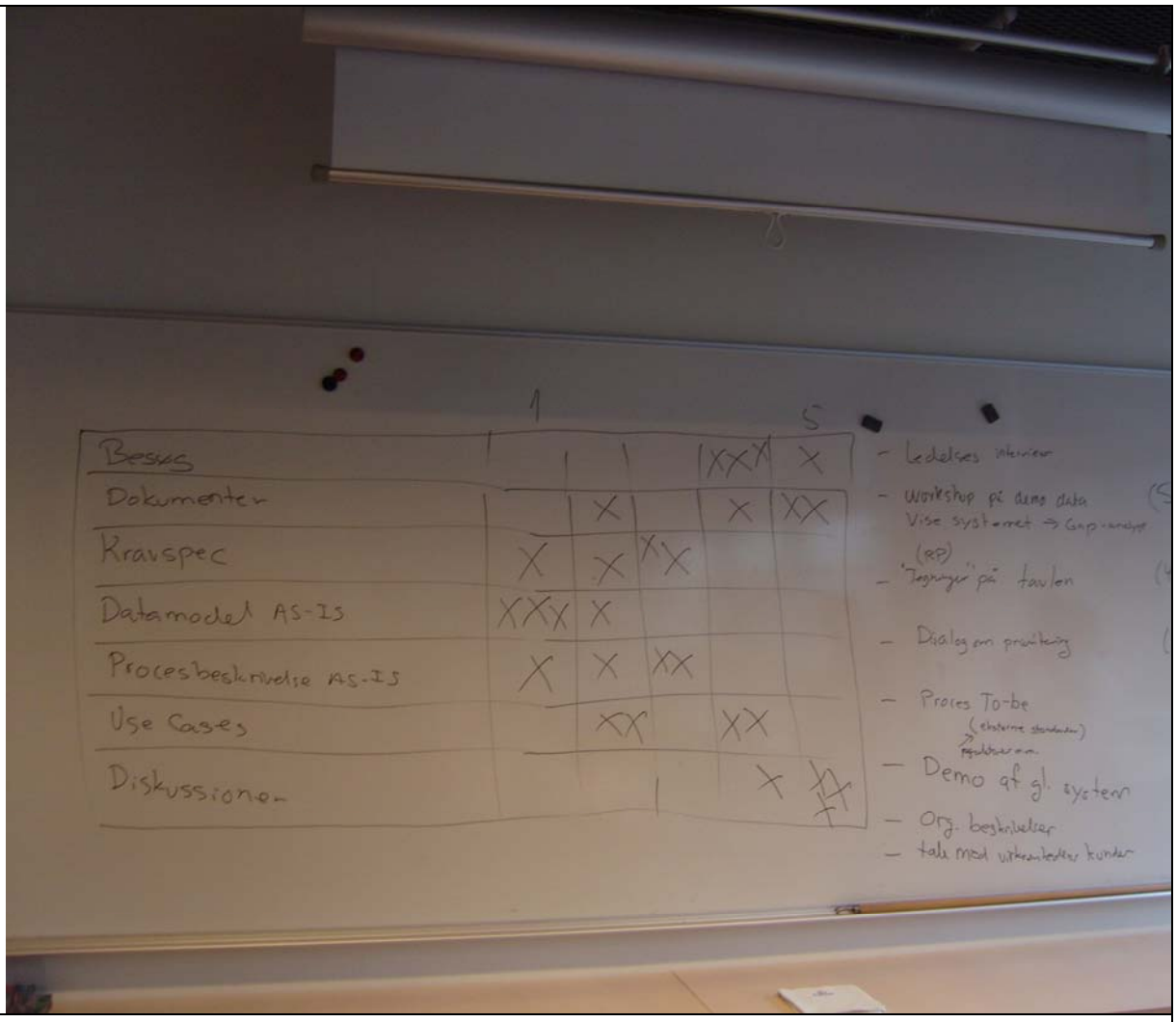
45	Lene: Jeg vil gerne skifte bane. Vi har et givet ERP system, vi har en given implementering. I forhold til den, i hvilket omfang er det så nødvendigt at lave en analyse af denne organisation og organisationens eksisterende processer i forhold til implementeringen? Jeg vil gerne bede jer diskutere indtil, det står klart for jer, hvorfor det er vigtigt eller ikke vigtigt.
46	Hanne: Inden man går i gang eller man har truffet beslutningen. (man har truffet beslutningen) så altså første del af projektet. Jeg synes jo det er meget vigtigt, nu har jeg jo også siddet på den anden side i lang tid. Det er om at forstå den forretning, som man skal levere sin løsning til. Det er essentielt for, om det bliver en succes i den anden ende. At man kan forstå processerne og forstå at mappe dem over i de løsninger, som findes i standard systemet.
47	Allan: Min holdning er, at man skal forstå det til en vis grad. Man er nød til at forstå virksomheden. Forstå deres kunder, deres produkter og deres leverandører. Deres medarbejders niveau og hvad det er, som gør den her virksomhed vigtig og unik. Således at man kan sørge for at holde fast i det. Og så helt overordnet, så skal man distancere sig så meget som mulig fra eksisterende processer. Ikke at man ikke skal forstå deres forretning, og man skal også forstå de administrative funktioner, som de har. Og så skal man tage udgangspunkt i det nye system, man må for guds skyld ikke bruge for meget tid på at forstå og beskrive as-is, men skal bruge meget tid på at forstå, hvad det nu er for en proces, som man forsøger at servere for virksomheden. Man skal ikke lytte til dem, som siger, at systemet skal tilpasses til virksomheden og ikke omvendt, dem skal man for guds skyld ikke lytte til. Man skal forklare og forklare og forklare, og man skal forstå hvor guldkornene ligger henne,

	man skal forstå, hvad det er der er vigtigt at holde fast i ud fra projektets målsætning. Man skal fokusere meget på to-be. Man bliver nød til at kradse i overfladen på as-is, man må endelig ikke bruge måneder på at lave analyser af eksisterende processer.
48	Hanne: Men man skal alligevel forstå essensen i den forretning, som der drives. Det synes jeg er vigtig, at man forstår.
49	Stig: De første år med ERP, der kan jeg huske, at de typiske processer, at man først skulle gå ind og se, hvordan alle de eksisterende processer gøres nu, for at de kan lægges over i, hvordan de fremtidige processer skulle være. Så gik der mange måneder, og man havde stadig ikke kikket specielt meget på systemet. I virkeligheden tror jeg ikke det handler så meget om, hvorvidt man kender forretningen, men at man får et rigtig godt team ind, og her skal der selvfølgelig både være nogle fra virksomheden, som kender processerne, men især også fra leverandør side, som kender systemet. Jo bedre team, der er der jo bedre succes får man.
50	Bjarne: Det er jeg helt enig i, også at man er nød til at vide noget om det sted, som virksomheden kommer fra. Fordi det giver mange af de bindinger, som en virksomhed mener de er nød til at have i den nye løsning. Bindinger fordi det var de nød til at leve med, fordi det er blevet så indgroet fra deres gamle vaner, at sådan er verden. Og den behøver slet ikke at være sådan, hvis man tager udgangspunkt i de linier der er i det nye system, de tanker der er i det nye system. Hvis ikke man er klar over det, at der er sådan en sammenhæng, så falder man let i at acceptere, at momskoden skal være en del af kontoplanen, selv om det ikke er en nødvendighed i den nye løsning eller andre tilsvarende. Som Stig siger, så bør man have et godt team der, det er enormt vigtigt, det er det vigtigste tidspunkt at have ikke mindst et meget erfarent team. Både erfarent fra leverandørens side omkring det nye system og omkring hvad man har gjort alle mulige andre steder, og i at kunne sætte sig ind i den nye kundes hverdag, verden, livsform. Tilsvarende fra kundens side. Dem der ved, hvorfor man gør som man gør, dem der ved, hvad der vil være bedst for vores virksomhed i fremtiden, hvor ligger der nogle benefits, som man kan samle op ved det her. Hvad er det, som koster os mange penge. Enten på grund af administrative omkostninger eller fordi det er for stort, det tager for lang tid, det er for usikkert eller vi når bare ikke at få det gjort eller hvad det nu kan være og dermed er vi for sent til at få de ordrer, som vi skal leve af, eller hvad det nu kan være for et område det nu drejer sig om. Det er vigtigt at sætte sig ind i det gamle systems virkemåde eller organisationens eksisterende forretningsgange på et overordnet niveau. Frem for alt for at vide, hvorfor er det lavet sådan. Og så er det enormt vigtigt at stille spørgsmålstegn ved det hele. Man skal ikke som konsulent acceptere, at i den her type der kan man bare ikke gøre anderledes. Det er en af de største faldgrupper, som en konsulent kan falde i, at lade være med virkelig at udfordre dem og virkelig at gå dem på klingen. Hvorfor skal det være sådan? Hvad med det her?
51	Hanne (og Bjarne): Jeg synes også det handler meget om, når man kommer ud, at se mønstre. Når nogen forklarer, at se hvordan forretningsprocesserne, at se de mønstre der er i de systemer, som man nu kommer fra, er genkendelige. (Bjarne: det er klart) Så kan det godt være, at de bruger nogle helt andre ord, at det bliver forklaret på en helt anden måde, men man kan ligesom omsætte det i forhold den løsning man (nu taler Bjarne og Hanne lidt i munden på hinanden) Bjarne: og det er en, det har noget med erfaring at gøre fra forskellige virksomheder (Hanne Ja…ja) og paralleller og selvfølgelig fra de system, som man skal

	have dem ind i. Det kan jeg også give et praktisk eksempel på.
52	Bjarne: Den gang jeg var hos Oracle, der var sælger, som var kommet til at sige på et møde med IT-chefen hos Sofus Berendsen i tekstil service delen, sådan en virksomhed så banal som jeres, hvad skal I overhoved med et ERP system. Det kan da ikke være nødvendigt, I skal jo bare vaske noget tøj og sørge for, at kunderne får det tilbage. Det er faktisk en rimelig kompliceret proces, nu man tænker sig om. Man skal samle det sammen hos nogle kunder, meget af det er jo med navn på f.eks., så det er den enkelte person hos kunden, som det drejer sig om. Det skal samles sammen og skal igennem en proces hjemme på vaskeriet, det skal ud igen og havne det rigtige sted. Og man skal sørge for, at den person der skal have en ren kittel har en at tage af hver gang det er nødvendigt at man skal have en ren kittel. Derfor er det faktisk rimeligt komplekst. Da vi så begyndte at tale lidt med dem om, hvad systemet skulle kunne og sådan noget, så var de jo sikre på, at Oracle havde diskvalificeret sig selv. Fordi en på et tidligere tidspunkt var der en, som havde sagt, at sådan en virksomhed havde nok ikke brug for et avanceret IT-system. Der prøvede jeg at drage paralleller til en anden virksomhed, som jeg havde arbejdet for. Det var en fuldstændig anderledes branche. Men der var nogle af de paralleller, hvor de sagde hov ja, sådan kan man jo også se på det, og så fik de pludselig mere indtryk af, at det var noget avanceret, at man skulle.
53	Lene: Det der synes at være tilfældet er, at I mener der er en eller anden balance, som skal findes, hvordan finder man den?
54	Hanne: Hvis det er mere et spørgsmål om, hvordan man gør tingene, så er man kommet for langt. Det er mere hvorfor og hvad er det, at der skal udføres, at man kikker på de her processer, på det niveau er det ok. Hvis vi er kommet over i hvordan, så er man meget løsningsorienteret, og det er vi slet ikke interesseret i. Man er egentlig lige glad med den kørende løsning. Så det er meget med, hvad skal der udføres og hvorfor.
55	Allan: Først og fremmest kommer det an på målet med systemet. Jeg kan huske at … jeg kan give et eksempel. Jeg skulle designe et system, som skulle godkende, frigive vare i biokemisk produktion. Det var Novo Syme, som skulle frigive deres insymer, så er der rigtig mange tests, som skal tages. Ph skal være i orden og der må ikke være for meget salmonella i og der var 35-50 tests. Der skulle vi lave et system, som automatisk godkendte så meget som mulig. Før sad der rigtig mange mennesker og kikkede på de der papirer fra laboratoriet, kontrollerede, satte et stempel på og tastede det ind i systemet, som så frigav varerne. Nu skulle vi lave det her system, som automatisk skulle lave det meste af matchningen, vi skulle have alle produktions data ind og alle laboratorium data ind og så skulle vi lave et system, som automatisk klargjorde så meget af godkendelsen som muligt baseret på grænse værdier. Så skulle man bare ind og kikke på skærmen og se, hvad computeren havde sagt og så sige ok på skærmen. Så virkelig man gik fra 30 mennesker og 25 dages behandlingstid til 6 mennesker og 2 dages godkendelse. Man arbejdede med at kikke tilbage på de tidligere batches, så man kunne få en hurtigere godkendelse. En rigtig god business case i at gennemføre i at gennemføre sådan et system. Der må man i hvert tilfælde ikke kikke for meget på arbejdsgangene, men vi blev nød til at forstå rigtig meget af de her laboratorium folks hverdag, for at bygge et rimeligt system. Dem der var med, de fleste var selvfølgelig ikke med, de vidste godt, at den her afdeling blev nedlagt. Men der havde vi virkelig at forstå meget af deres virkelighed. Med hvordan man godkender de her batches og var nød til at arbejde meget med processerne. For overhoved at arbejde med det, men det var ikke så

	meget ….. vi kikkede forfærdelig lidt på den der proces med, hvordan de samlede alle de her ting sammen, og gav dem videre, der var også nogen, som prøvede at implementere det der med forsegling også skulle med, og numre på de forskellige laboratorier, det kunne man også godt lave i systemet, men det kæmpede vi en lang sej kamp med slet ikke var nødvendigt længere at lave de er nummereringer af forskellige laboratorium batches. Vi kunne bare tage de rå resultater og hælde dem direkte ind i systemet. Det kæmpede vi en lang kamp imod, og der havde vi i hvert tilfælde ikke noget behov for at forstå det der mere grundigt. Det er sådan et eksempel.
56	Allan: Et andet eksempel var hvor vi skulle implementere et standard system i 19 lande. Det helt traditionelle, økonomi, salg og produktion. Der havde man ikke nogen changse for eller ønske om at forstå alle de lokale processer. Der viste man, at der jo virkelig var grobund for at lave 19 forskellige systemer i 19 forskellige lande. Der brugte vi meget lidt krudt på at forstå, som udgangspunkt, at forstå processerne. Der tog vi tre lande som ”gidsler” kunne man sige, på en pæn måde, men tre lande som gidsler var med inde og sige, kan I leve med de her standard processer? Så sagde de, det tror vi godt vi kan, og så sagde vi, når I kan så kan resten af landene også.
57	Stig: Jeg kan give et andet eksempel, hvor man også tog gidsler fra de forskellige lande og brugte dem til at tage rundt og få implementerede flere selskaber også, og vi skulle også bruge nogle på hovedkontoret. Formålet var i virkeligheden at finde ud af, hvad lavede de egentlig i datterselskaberne. For når man så havde fået rullet tingene ud, så var formålet at få lavet et chared service center og dermed mere eller mindre nedlægge funktioner der ude. (Allan: det var i virkeligheden det samme), men hvor man bliver nød til at implementere det, for at finde ud af, hvad de egentlig lavede der ude, det havde man egentlig ikke styr på. Det er jo nok den største achileshøl. Hvad laver sådanne nogle datter selskaber i virkeligheden? De laver det meget besværligt, men det føler de, at de er nød til, fordi de skal kunne dokumentere det på et andet niveau end moderselskaber.
58	Hanne: En af de ting, som jeg synes er fordelen ved et ERP system, hvis man få det implementeret hele vejen rundt, det er, at det skaber en gennemsigtighed. Også på tværs af nogle afdelinger, som måske ikke er så hensigtsmæssige. Det kan godt være, at man tror at man viste hvad man lavede før, men når du først har et system, hvor du har den der integration, så bliver tingene altså meget mere synlige. Og så bliver der mulighed for at kunne lave nogle af de her optimeringer bagefter.
59	Stig: Men det er også det, som er farligst når landeledelserne nu ikke mere kan holde tingene skjult 3-6 måneder. Når moderselskabet finder ud af, hvad det egentlig er de har gang i. Nu bliver det måske opdaget indenfor samme uge eller samme måned.
60	Allan: En af de globale implementeringer jeg var med til, det var også det der med, at først så implementerede man egentlig as-is, men så med standard processer. Så fjernede man lagerne fra landene, det var en af deres hjerte børn, og så fjernede man hele administrationen bagefter så det kun var et salgskontor. Det er en meget effektiv proces. Man sparer virkelig mange penge, ved at gennemføre det på den måde. Der var det også meget sjovt, for der var vi ude i hver enkelt land efter vi havde lavet den her standard proces. Så havde vi den her kamp i alle lande, hvor de sagde ”de lovgivningsmæssige krav”, dem vil vi gerne implementere. Så var der den her lange kamp med, at det her er et lovgivningsmæssigt krav.

	Vi implementerede i England og Tyskland o.s.v. og det tog lang tid, jeg ved ikke om vi fik pillet dem alle sammen fra, men der brugte vi erfaringerne fra andre virksomheder. Ok har I implementeret det her hos andre, de siger det er et lovgivningsmæssigt krav i Tyskland. Og så ringede vi til en anden Tysk virksomhed og spurgte, har I implementeret den her proces? Nej det har vi ikke. Hvordan kan I undgå det, hvis det er et lovgivningsmæssigt krav? Den vej var vi nød til at gå. Det var kun da vi kom til Italien, hvor det var os der fik lovt til … der var det den modsatte proces. De stillede alle mulige obskure krav til systemet. Så sagde vi, det kan ikke passe og det kan ikke passe kan ikke passe, at vi skal lave det, indtil vi ringede til et par andre virksomheder. Jo, jo den er god nok, det er et lov krav (alle griner rigtig meget – som om det har de også prøvet). Der var det os der lærte det.
61	Stig: I det projekt jeg fortalte om, der brugte de den metode, hvem kommer i fængsel, og er det mere end 6 måneder? (der grines igen meget – Allan siger, ok den er hård nu)
62	Lene: Vi har nu prøvet at diskutere, hvor finder man balancen, og noget eksakt svar får vi nok ikke i dag, men en eller anden form for information om organisationen skal man have og det jeg har lavet her er en liste over mulige måder at få informationen på, og jeg har også lavet en box med andet, og den er I meget velkommen til at skrive i, der er sikkert mange andre måder at indsamle information på.
	(De udfylder skemaet og jeg laver en skitse på tavlen, hvor deres resultater kan noteres, så alle kan se dem).
	Lene: Skal vi starte med at se, om I er enige i dem jeg havde listet.
63	Bjarne: Der er himmelvid forskel på, hvad det er man skal implementere (Hanne: Ja). På, hvor vigtige de her forskellige ting er, og hvad det samlede projekt går ud på. F.eks. er der en data konvertering med eller ej i projektet. Det har en hel del at gøre med, om vi skal bruge datamodellen as-is.
64	Lene: Jeg plejer selv at sige, at der ikke findes nogen standard beskrivelse af en implementering, Men hvis vi tager en generel implementering i en organisation, hvor det er de fleste moduler, som de skal have på en gang. Sådan et traditionelt ERP implementering. Så lad os lyn hurtigt se, hvad I har svaret.

	
65	Hanne: 5, 2, 3, 1, 1, så har jeg ikke sat noget på use cases fordi jeg ikke rigtig, for mig er det en end-to-end beskrivelse af en bruger dialog. Det er ikke noget, som jeg overhovedet har arbejdet med. Men jeg vil give den 2, og diskussion 3?.
66	Stig: 4, 2, 1, 3, 4 og 4.

Appendix B section 2

Tidsplan for ERP fokusgruppe torsdag den 15. november 2007

Tid	Spørgsmål	Materialer
17.00	Kaffe, vand, frugt og introduktion Spørgsmålene før pausen er mere generelle og har til formål dels at be-/afkræfte nogle af de mere generelle problemstillinger, som jeg fandt i case organisationerne. Efter en lidt længere pause ca. kl. 19 Vil jeg præsentere et lille scenario for jer, om illustrer centrale problemstillinger, som jeg fandt i mit case studie.	- Præsentation af mig og mit projekt - Præsentation af de enkelte deltagere - Formålet med fokusgrupperne - Hvordan får de feedback - "Reglerne" for fokusgruppeprocessen o Facilitering o Film o båndoptager -
17.15	**Spørgsmål 1** Hvad er efter jeres opfattelse de største udfordringer i forbindelse med implementering af ERP systemer? I hvilket omfang er det nødvendigt at analysere organisationens eksisterende arbejdspraksis? (og hvad er formålet med analysen?) I skal diskutere indtil det er klart for jer, hvad der er vigtigt og hvad der ikke er vigtigt	Det første spørgsmål er blot til at åbne processen med. Er der reelt forskellige muligheder for design af ERP-systemet – hvor forskellige kan implementeringerne blive?
18.00	**Spørgsmål 2** Hvor vigtigt er det for implementeringskonsulenterne at forstå den aktuelle brugerorganisation i forbindelse med design (konfigurering og definition customizeringer) af det nye system? Hvilke aspekter af den eksisterende organisation er det nødvendig at forstå – hvad er relevant viden?	- Skema til deltagerne - Tavle anvendes til at summere og diskutere resultatet

	Har det betydning, at konsulenten har erfaring relateret til domænet – har samme uddannelsesmæssig baggrund som brugerne, har arbejdet i en organisation med det samme arbejdsområde eller måske blot implementeret i mange lignende organisationer? Hvor vigtig er følgende (på en skala fra 1-5) i forhold til at give konsulenterne relevant viden om brugerorganisationen? - Besøg i forskellige afdelinger for at se dem arbejde. - Dokumenter (ordre, faktura, procedure beskrivelser m.m.) der anvendes i den eksisterende organisation. - Kravspecifikation udarbejdet af organisationen. - Datamodel af AS-IS - Proces beskrivelser af AS-IS. - USE CASES - Diskussioner med brugerorganisationens repræsentanter i forbindelse med design (konfigurering og ændrings specifikation af systemet). - Andre måder at opnå viden på. (Hvad forstår de ved relevant viden?)	
18.30	**Spørgsmål 3** Er det vigtigt for brugerrepræsentanterne at forstå standard ERP systemet i forhold til at deltage i udarbejdelsen af designforslag for det nye system? - I skal diskutere problemstillingen indtil det er kart for jer, hvorfor i mener det er vigtigt eller ikke vigtigt! Hvor vigtig er følgende (på en skala fra 1-5) i forhold til at give brugerrepræsentanterne relevant viden om standard ERP systemet? - Uddannelse i standard systemet - Udforskning af standard systemet på egen hånd - Gennemgang af procesmodeller	- Skema 2 - Skrive på tavlen

	(f.eks. en reference model) - At sidde sammen med konsulenten og se forskellige skærmbilleder - Diskussioner og udforskning af systemet sammen med konsulenterne - Andre måder at opnå viden på (Hvad forstår de ved relevant viden?)	
	Pause	
19.30	**Spørgsmål 4** Jeg vil gerne fokusere på evaluering af løsningsforslaget nu. Hvor vigtigt er det, at forskellige brugergrupper og konsulenterne har et fælles "sprog" til at diskutere løsningsforslag? - I skal forsætte diskussionen indtil det er klart for jer, hvorfor I mener det er vigtigt eller ikke vigtigt! Hvor god er følgende teknik (på en skala fra 1-5) i forhold til at beskrive et løsningsforslag så brugerrepræsentanten kan forstå og diskutere det? - En løsningsbeskrivelse illustreret på en tavle - En løsningsbeskrivelse i et dokument med f.eks. data model og tegninger - En procesbeskrivelse der tager udgangspunkt i ERP systemets funktionalitet (f.eks. work-flow format) - Konsulenten gennemgår den relevante sekvens af skærmbilleder - Udforskning på egen hånd af løsningen i et testsystem - Andre måder? Hvordan håndteres definition af stamdata, kategorier m.m.? - Giver det evt. anledning til konflikter? I skal blive enige om en prioritering af dem	Skema 3 Posted notes som deltagerne kan skriver på, og siden kan de sættes op på tavlen

	der er foreslået	
20.45	Afrunding og evaluering	- Hvad har de synes om processen? - Hvordan og hvornår får de et resultat - En lille gave til dem

Appendix B section 3

Skema 1

	1 (ikke vigtig)	2 (mindre vigtig)	3 (har nogen betydning)	4 (vigtig)	5 (meget vigtig)
Hvor vigtig er følgende (på en skala fra 1-5) i forhold til at give konsulenterne relevant viden om brugerorganisationen?					
- Besøg i forskellige afdelinger for at se dem arbejde.					
- Dokumenter (ordre, faktura, procedure beskrivelser m.m.) der anvendes i den eksisterende organisation.					
- Kravspecifikation udarbejdet af organisationen.					
- Datamodel af AS-IS					
- Proces beskrivelser af AS-IS.					
- USE CASES					
- Diskussioner med brugerorganisationens repræsentanter i forbindelse med design (konfigurering og ændrings specifikation af systemet).					
- Andre måder at opnå viden på.					

Skema 2

	1 (ikke vigtig)	2 (mindre vigtig)	3 (har nogen betydning)	4 (vigtig)	5 (meget vigtig)
Hvor vigtig er følgende (på en skala fra 1-5) i forhold til at give brugerrepræsentanterne relevant viden om standard ERP systemet?					
- Uddannelse i standard systemet					
- Udforskning af standard systemet på egen hånd - Gennemgang af procesmodeller (f.eks. en reference model)					
- At sidde sammen med konsulenten og se forskellige skærmbilleder					
- Diskussioner og udforskning af systemet sammen med konsulenterne					
- Andre måder at opnå viden på					

Skema 3

	1 (ikke vigtig)	2 (mindre vigtig)	3 (har nogen betydning)	4 (vigtig)	5 (meget vigtig)
Hvor god er følgende teknik (på en skala fra 1-5) i forhold til at beskrive et løsningsforslag så brugerrepræsentanten kan forstå og diskutere det?					
- En løsningsbeskrivelse illustreret på en tavle					
- En løsningsbeskrivelse i et dokument med f.eks. data model og tegninger					
- En procesbeskrivelse der tager udgangspunkt i ERP systemets funktionalitet (f.eks. work-flow format)					
- Konsulenten gennemgår den relevante sekvens af skærmbilleder					
- Udforskning på egen hånd af løsningen i et testsystem					
- Andre måder?					

Appendix B section 4

Metaphor	Perception of the IT-consultant's role	Perception of the system's role	Goal for the implementation	Perception of the role of management	Perception of the role of design team	Perception of the end users
A standardization war	• Defending the (IT-) system • Working for standards (management)	• Best Practice (standard) • Common infrastructure	• Integration on all levels • Visibility (data and business processes) • Common, operation and maintenance of the IT-system	• Deciding on the ERP system • Defending the standard	• Configuring an organizational wide standard IT-system	• Adapting to the standard • Passive – receiving training in the new IT-system
A game	• Serving a specific stakeholder in the organization (typically the IT-department) • Recognize stakeholders on both side change over time ➔ the game changes over time • Guarding the standard system	• Supportive infrastructure (hosted by IT-department) • Cost effective IT-solution • Work for all organizations	• Stakeholder satisfaction • Minimize cost (and effort) developing and maintaining IT	• An important stakeholder • Sponsor • Court of appeal regarding design decisions and conflicts between stakeholders • Support standard IT-system for cost reasons	• Negotiating design suggestions • Focus on the IT-artifact	• A stakeholder • Able to directly or indirectly influence the design team and the final design • Able to influence other stakeholders • Receive training in the new IT-system
A change project	• Working for the work design team "experts" (process designers) • Develop design suggestions based on the business case/requirements specification deploying the standard system as much as possible	• Designing the IT-artifact is secondary to the change process • As much reuse (standard) as possible	• Changing the way the organization work • IT-support to fulfill the business case • Integrated and optimized processes	• Develop or approve business cases (ensure strategic fit) • Communicate why change is important • Ensure resources for the project	• Strategic focus (IT-artifact secondary) • Design experts able to optimize the business and work processes (could be either internal or external experts) • The design teams work can be guided by the business cases	• Informants • Need to be included for psychological reasons and to some extend for knowledge diffusion • Receive training in new (socio-technical) processes

For each of the metaphors I have gone through the transcripts to find sections that support my perception of how the stakeholders role.

A standardization war	Citations
Arguments for the metaphor:	"*It is much about standardization and integration …….. You get into trouble if you do not win the* ***standardization war****. …..… Having everybody use the same system is both the challenge and the goal*" (focus group section 18) "If you come from a tailor made system you are sure to be **defeated**." (focus group 1 section 28) "*Here we took three countries hostage you could say, in a nice way, but three countries were* ***taken hostage*** *having to answer if it would be possible to live with these standard processes. If they said: we think so, then we said: If you can then the others can to. ….They are specialists within their domain but not necessary strong enough to convince the consultant that the "standard" is not working for us here."* (Focus group 1 section 56) Also on the organizational side the war metaphor is recognized. *"I have seen consultants* ***take my user representatives as hostages****. I know it can sound sharp but some of them are not use to or tough enough to go up against consultants.*" (Focus group 2 section 241)
The perception of the IT-consultant's role:	"Reuse also has something to do with costs. Previously we talked about the board of directors; we are all focusing on the bottom-line one way or another." ((S) focus group 1 section 42) **Working for or on behalf of management:** "*I can give another example where we took hostages from different countries and used them*

	to tour and implement in the subsidiary companies. The objective was to be able to see what was going on in the different subsidiaries. The final objective was to implement a shared service center, and close the local centers." (Focus group 1 section 57a) (A): "*Actually it was the same in my example, but it was necessary to implement the system to be able to see what they actually did out there. No one knew! It is a bit of a mystery; what do such subsidiaries actually do? They are doing things in a very inefficient way but the parent company fill they have to document it.* (Focus group 1 section 57b)
Perception of the system's role:	"*It is about providing common master data*." (Focus group 1 section 33) "*Maybe you thought you knew what you did before, but when you get the new system with the cross organizational integration then everything gets much more visible. Then it becomes possible to make optimizations afterward*." (Focus group 1 section 58) "*One of the global implementations I participated in, it was about implementing as-is first with standard processes. Then local warehouses were removed, which were one of the local darlings, and finally all administration was removed leaving only the local sales offices. It was a very effective process. They really saved a lot of money*." (Focus group 1 section 60) ➔ also have implications about the goal of the implementation.
Goal for the implementation:	"*You want one integrated system sharing master data, using the same processes and input screens…..it is all about standardization and integration. No one think that is fun*" (focus group 1 section 18)
Perception of the role of management:	They ask for a standard and are defending the standard
Perception of the role of the design team	Helping implement a standard
Perception of the end user:	"No organization want the end users customizing the system" (Focus group 1 section 23)

	"*In each country we had a fight over customizations needed because of local legislation. It was a long fight, and we used experience from other companies implementing the system in the same countries. We simply asked other consultants implementing in the same country if they had to make a customization because of local legislation, if not …. It took a long time to document it was not necessary to make the customizations they asked for. Italy was an interesting exception, all they asked for we had to make. There **we** learned a lot*." (focus group 1 section 60) ➔ the perception of the user is, that they need to be taught a lot, and that they are cheating.
	"As part of the companies group management, the situation was, that we had some subsidiaries that needed to have the system **enforced on them**" (Focus group 1 section 10)

A Game	Citations
Arguments for the metaphor:	"*I think this is a problem of grate important in this **game***" (Focus group 1 section 17)
The perception of the IT-consultant's role:	In (focus group 1 section 17) different stakeholders over time in the project is recognized. "M*any different goals also on the vendor side*"; sales person, pr-sales consultant, implementation consultant, programmer, support department and after sales."
Perception of the system's role:	Supportive infrastructure. Using the IT department's perspective, perceiving the system as servicing the organization. Deliver best practice at a reasonable price. "*One of the more important things about the large systems is the enormous customer base. Over the years experience is collected, thus not just one way to do things but a number of variants. Meaning that there should always be a solution which any company can live with, or at least use and find right for them."* ((B) focus group 1 section 41)
Goal for the implementation:	"*For who; the customer or the vendor? For the vendor it is about selling licenses. …… at the end it is all about earning some money. For the customer it is to get a solution to some basic problems.*" (Focus group 1 section 16). *"At the customer side you have equally diverse goals. It can be one person who wants to thumbprint something. Obviously the person believes it is for the best of the company, but it is important for him that he made it happen. In the organization there may be others with completely different goals, and maybe somebody that actually will suffer from the decision because the group participating in the decision or the considerations taken into account when taking the decision was to narrow. …….… It gets more and more impossible to oversee the consequences of such a project" (Focus group 1 section 17)*
Perception of the role of management:	An stakeholder primary seen as the sponsor of

	the project. Included in the steering committee having the roll as “court of appeal” in relation to decisions and conflicts involving more stakeholders.
Perception of the role of the design team	Consist of representatives from internal stakeholders who negotiate the design. This is what characterizes a real consultant; he will not just give the organization what they ask, it is a dialogue where both parties have to accept, that the business (users) cannot claim that business conditions require you to do things in a specific way. They have to explain what is needed and then we look at the possible solutions deciding what is better. Then you get a feeling for the standard and then you follow it through. Obviously you should not just implement standard everywhere, but customizations are expensive and they make upgrading a nightmare. What you need to figure out is where it is financially sound to be special.
Perception of the end user:	Stakeholder. Individual groups or coalitions influence the design.
	“Basically you have to accept ….you are having something new (an ERP system), but at a matter of facts you don’t know what you get. ……you need someone that is willing to take a risk to travel with you on the journey.” (Focus group 2 section 50)

A change project	Citations
Arguments for the metaphor:	"*At XX I participated in a "2000" implementation of SAP, many years later we did a re-implementation starting all over, not because we were wrong the first time, but the business had changed and new modules had been added to SAP*." (Focus group 1 section 30) *"It is important to approach it as a* ***change project*** *that is the major challenge. If you approach it as a business project then very different mechanisms come into play from the very beginning. It creates a very different communication with the organization. The expatiation you create in the organization is different compared to perceiving it simply as an implementation of an IT system. It is very, very different."* (Focus group 2 section 21)
The perception of the IT-consultant's role:	**Work for the design team:** "*You need external resources for two things: as resources in areas where you don't have sufficient internal work power or for key competence that you lack in the organization. If I have to be very clear, that is their role*." (Focus group 2 section 68) **Develop design suggestions based on business cases:** "*With vendor XX the mission was to define the business case and follow it through (force it through the organization) ……. The perception was that we (us as consultants) acted as an auxiliary arm to management*."(focus group 1 section 20) **More general:** "….having someone internal in the organization, someone as me, having been an external consultant previously makes it possible for us (the organization) to ask the critical questions on behalf of the organization. Because we know where the holes are. But if you don't have the competence internally you could try buying the competence externally. You might

	buy someone to challenge you on the requirement specification or the project objectives. The requirements specification should reflect the goal, the vision and the strategy to get there but it is not always the case. You can find someone to challenge you on that but it takes time ….” (Focus group 2 section 75)
Perception of the system’s role:	Secondary to the goal for the change process.
Goal for the implementation:	Business goals.
Perception of the role of management:	“*The steering committee should have the right composition so that they can follow the change through in the organization. They have to show up when the music begins* (complaints and resistance to change, my addition). *The worst that can happen is that you don’t get appropriate support from the sponsor; if they aren’t living up to their promises. You need a clear business case as a lever to drive it through the organization. So that you have something to knock it into the organization.* (Focus group 2 section 21) “I agree with (H) that if you consider it a business project (a change project), but …..(pause) if you are implementing something new it is by nature a change ….(pause) but what you say is, that top management has to work with the banner …..(pause) give it much attention communicating; arguing downward in the organization, explaining what is needed and why. (Focus group 2 section 26) “Often top management say something has to happen, but people at lower levels don’t know what to do or what is the plan and the objectives” (Focus group 2 section 27)
Perception of the role of the design team	“If it is just a matter of changing one IT system for another, then it is one type of project. If it is suppose to change and develop the business then it is much process oriented. You have to be open to what kind of project…. It is very common the organizations have fear of contact, they won’t face the consequences. They would like the result, but it comes with a cost. It is here the business case comes in; what do we gain and what will it cost? …. You have to know it and plan accordingly! “ (Focus group 2 section 24)

	"*In my experience IT-involvement (IT knowledge) is often lacking when describing business requirements. Sometimes business requirements are made based on a previous system just adding some whishes. Business consultants representing smaller areas cannot oversee the consequences of implementing a standard system covering the entire organization. All they do is thinking about their own area and require what will make them stand well with their own group. But it is an integrated standard system that we are implementing, they don't understand that data entered in one module is made available elsewhere. This kind of relations IT people has to help explain, plus cut down business requirements that come out of the blue. The way you have to act is questioning everything all the time. Should this really be included? Are you sure this is necessary? What do you need this for? Can't it be done in a different way? I have seen many business people requiring what is easy for them to get acceptance for in their hinterland."* (Focus group 2 section 25) "*Regarding responsibilities and challenges I don't think the organization can expect "a brilliant consultant" to take over the responsibility or require them to give the right advice. It is the organization's own responsibility. If you are not aware of your own responsibilities then you make a big mistake, and excuse me, but then you binge it on yourself if you get the wrong consultants. You should rather get the right employees in key roles; have them ask the right questions, someone that know the marked and understand both IT and the structure; both advantages and disadvantages. Someone that have all-round knowledge and are able to make the connection to the business."* ((H) Focus group 2 section 57) "*It is my experience, that if you are the kind of organization that needs a tender process, then it takes time, maturing time; attitude time I might say. Top management is involved and you are turning many stones. You get into all corners*

	and hopefully you have time to consider it thoroughly. You also have time to make a risk profile for your project, so that you start the project with your eyes open and know how to do it. Instead of starting the project to quick, because you want to finish soon and then shooting yourself in one foot. Becoming a project manager in that situation is not good, because you get into discussions that actually should have been take upfront while making a thorough analysis including a business case. That is very important to have it done that way." (focus group 2 section 59) "*….the time all together is the same….* " (Focus group 2 section 61" Response to (J) objection that you don't have time.
Perception of the end user:	"You need organizational development to succeed!" ((H) focus group 1 section 28) "There is a saying; it isn't until the third implementation ERP is a success – which you can think about as a consultant if you are doing a second implementation (much laugh among the participants)! ((S) focus group 1 section 29) "You have to figure out how your organization should work after the implementation, all which is necessary to make it work. How you make the change come true is important." (Focus group 2 section 28) ➔(H) on one side and (S+E) on the other side get into an argument if perceiving it as a change project and focus on the organization is more important than making sure that requirements specification, test organization and so on is in place. (S+E) argue that resources are always a problem and that traditional IS development tasks are important. (Focus group 2 section 13-31)

Issues cutting across the metaphors:

Conflict between consultant and organization	
Internal versus external consultant	"*Working internally the organization say: the system shall not dictate how we shall work! Thus it is difficult to mutually adapt expectations. So it is perceived as something positive by the organization. I have only been an external external consultant for a year [worked internally for many years]……I think it makes a grate difference if you are internal or external consultant when you have to deliver the solution. ….If you have to live with the solution for many years to come versus leaving the organization through the backdoor after go-live makes a difference when making decisions because the agenda is different. It is fair enough, but it is two very different things I think.* " ((H) Focus group 1 section 19) ***"All the things you as a consultant think are stupid or the things organizations don't do; now you suddenly can see how difficult it is to make it work."* ((S) Focus group 1 section 11)**
Different vendors work for different stakeholders	Mærsk: It department Accenture, McKenzie: Management IBM: for It department (Focus group 1 section 20)
Users/organizations not being able to make rational decisions	"customer XX used 11 segments in their chart of accounts, one of them the VAT code …. Basically this destroys the idea of a standard system. (Focus group 1 section 24) "I also experience a wish to map the old world to the new. Often because your had no other way to handle the complexity in your old world than to build it into the chart of accounts. Now you have a large application portfolio providing functionality but……." (Focus group 1 section 27) "*They try to customize the system as much as possible to the organizations way of working instead of living with the more limited possibilities the standard system provides. Suddenly it is no longer a standard system."* (Focus group 1 section 23) - Both (A) and (B)

	continue the discussion on how the borderline for customizations has changed over time, but also talk about that it has to be managed/controlled very tight. ➔ **My conclusion the design space has been enlarged over time.**
Conflicts between different stakeholders	(Focus group 1 section 17) Many different stakeholders on vendor side as well as on customer side. *"At the customer side you have equally diverse goals. It can be one person who wants to thumbprint something. Obviously the person believes it is for the best of the company, but it is important for him that he made it happen. In the organization there may be others with completely different goals, and maybe somebody that actually will suffer from the decision because the group participating in the decision or the considerations taken into account when taking the decision was to narrow. …… It gets more and more impossible to oversee the consequences of such a project" (Focus group 1 section 17)*
Consultants against users	Question 1 from me: What is in your opinion the major challenges implementing ERP? "*The users ….. (some laughing from the others) I don't think you can be more precise …."* (Focus group 2 section 11) "*Use standards don't develop different obscure corners to the system because the user organization claims their life depend on it*." (Focus group 2 section 13) - (S) and (E) confirm each other in this. In focus group 2 the opening remarks set the stage for the arguments between user representatives and consultants. It seems to be a continuation of the fight taking place in the ERP projects. "I understand when users say this customization is just a small thing. When you in a browser just

	need one click to make a short cut and jump 6 screens to get the functionality you would like, why can't I get it in SAP?" (Focus group 2 section 54) "Because he needs to test it every three months, when you upgrade or make other changes. It will cost so much, and at the end he will be tired of it. But as a user you cannot oversee it if you have no experience with SAP, and that is often the case for the business representatives." (Focus group 2 section 55) "……it is only in countries with high salary that usability is interesting. If we can save 10% of the time used on something then we save much salary. In other countries in e.g. Asia saving time/salary is not as important. If the user interface is cumbersome then another employee is just hired. The price of entering data is low compared to making customizations." (Focus group 2 section 56)
The nature of an implementation	"*For me one of the main issues is that those involved in preparing and signing the contract is to distant from fulfilling the contract and actually taking place in the work. This is very complex, and it gets more and more complex every year. The technology provides more and more opportunities thus it gets more and more impossible to oversee the consequences of such a project*." (Focus group 1 section 17)

Design of coordination/integration	"*Integration is build into a standard system. If you follow the processes imagined* [*prescribed by the system;* my addition]. *Thus the point is not whether or not it is easy to do customizations. The problem is that if you make customizations it is very difficult to oversee the consequences of this customizations in an integrated environment*." (Focus group 1 section 32) "The system itself is a standard system, but within the system you have the choice of many different standard work processes e.g. within purchase." ((S) Focus group 1 section 35)

	"For me the huge customer base is important. Through the years experience is picked up and therefore not just one but a number of variants has been incorporated within a given area. Thus actually there should always be a possible solution, that each and every one company could live with; one they can use and should find right for them." ((B)Focus group 1 section 41) "…*A (mellemmand) with a foot in each camp, a bridge builder, having knowledge about the system and the business (organization). A kind of translator*." (Focus group 2 section 76) (E) Agrees in the back ground. "*In my department we have 25-75% profiles. 75% of the knowledge and focus in on the HR domain and 25% on IT. For internal and external consultants we strive for the opposite. …. Thus you have different focus areas but something in common. It is the key to success. ….. you need something to be able to communicate*" (Focus group 2 section 77 - 81) "When you start an ERP project then the organizational representatives know 75% of their business and 25% of the old system, and the internal consultants can through away 75% of their knowledge related to the old system. In come some SAP consultants that have 100% SAP knowledge, but don't know the actual business. You need the common ground; I agree. You may be able to have someone with 100% business knowledge, but only if you don't have someone providing the common ground on both sides! …." (Focus group 2 section 84) "It is possible to meet the good consultant who has it." (Focus group 2 section 85) "Or the organizational representative who has it. The problem is; you don't have it the first time." (Focus group 2 section 86) "*That is the case for all customers and it is a problem right form start. In the tender process they don't have the knowledge to evaluate if the*

	system is good enough. I think it is sad that you don't think about it already from this point in the process." (Focus group 2 section 87) They have a short discussion if it is possible to get consultants that worked for your competitor. ….. *"You actually expect experienced consultant with knowledge about the line of business that can be reused when you deal with standard systems. But then where is the strategic advantage? ….I can look back on some successful projects where the customer had no prior knowledge about SAP. It is the change management process that went right because it implies the right communication and the right tools, and then it is less important what you know at the beginning.*" (Focus group 2 section 91) "It is about the organizational attitude; sometimes they are not willing to compromise. (Focus group 2 section 92) *"It is still about the change process"* (Focus group 2 section 93)
The consultants role	*"It is very important to question all information you get about the existing organization. You should not accept that in an organization like this it can't be any different. It is one of the largest pitfalls you as a consultant can fall into; not truly challenging them (the organizational participants), actually not pressing them hard. Why should it be like this? What about this here?" (Focus group 1 section 50)*
What do you mean by STANDARD system?	"*A standard is when you don't customize the system. In an ERP system you can do much without programming."* **SAP perspective** (focus group 1 section 22) …… I like your idea of a shared database; maybe you could take it to an even more abstract level. A standard system is when you accept and follow the work processes as they are imagined by the vendor (designer of the system). (Focus group 1 section 27) "*The borderland has changed, now you can do a lot without programming using different tools". I would say it is a standard system if you buy the "box" from a specific vendor, use the main part of the functionality and it is able to*

	cover the majority of your business needs." **Oracle perspective** (Focus group 1 section 23) "To me a standard system is only a frame which contains something, which is more or less customizable, allowing you to adapt or adjust." (Focus group 1 section 28) "Previously ERP was about executing now it is about planning, executing and follow up. Before executing and follow up was split into different systems. (Focus group 1 section 31) "It is about providing common master data." (Focus group 1 section 33) **Statement 36 and 38 from (J) are somehow contradicting – showing a dilemma**. "*I think it is about managing expectations because as a customer we do not accept to be wrapped up in a standard system; it is an insult to the company which relay on distinctive competences, something that our competitors do not have*." ((J) focus group 2 section 36) "*Basically I agree it is about getting the cheapest implementation; the most effective utilization of the standard components that the system contains. That is where the money is. Those who earn money are those that understand how to utilize the standard system not customizing it too much.*" ((J) Focus group 2 section 38) (H) add to the discussion: "I think it is a truth with some modifications. The way SAP is now it is very hard to distinguish what is standard and not standard. It is possible to place almost anything under the standard." (Focus group 2 section 39) ➔ (S) Again defend the argument (E back her up), that the system should not be modified, and that the customer will get into all sort of problems and cost issues. (H) Argue back that WEP portal allow you to change the process

	flow but still use the standard components, it is even suppose to be used that way! (E) give the example that first you use five years implementing everything in SAP and then you realize that it is just as complex as before and then start a process cleaning up the system returning to something close to standard. "**This is what characterizes a real consultant**; he will not just give the organization what they ask, it is a dialogue where both parties have to accept, that the business (users) cannot claim that business conditions require you to do things in a specific way. They have to explain what is needed and then we look at the possible solutions deciding what is better. Then you get a feeling for the standard and then you follow it through. Obviously you should not just implement standard everywhere, but customizations are expensive and they make upgrading a nightmare. What you need to figure out is where it is financially sound to be special. It is properly not within finance, we have to find out where; purchase is another area you wouldn't customize, HR properly not. (Focus group 2 section 51) ➔**Most of the discussion in focus group 2 is about the conflict between consultant and users, and for or against customizations. Can the organization live with standard or not!**
Is it about reuse? (standardization)	"*If you understand it as something that mature over time and get some kind of - or are based on a kind of best practice. But it is still about reuse. One of the more important things about the large systems is the enormous customer base. Over the years experience is collected, thus not just one way to do things but a number of variants. Meaning that there should always be a solution which any company can live with, or at least use and find right for them."* ((B) focus group 1 section 41) "Reuse also has something to do with costs. Previously we talked about the board of directors; we are all focusing on the bottom-line one way or another." ((S) focus group 1 section 42)