

2007

Making Sense of Project Management - A Case of Knowledge Management in Software Development

Annemette Kjaergaard
Copenhagen Business School, amk.inf@cbs.dk

Karl Kautz
Copenhagen Business School, karlheinz.kautz@rmit.edu.au

Peter A. Nielsen
University of Aalborg, Denmark, pan@cs.aau.dk

Follow this and additional works at: <http://aisel.aisnet.org/ecis2007>

Recommended Citation

Kjaergaard, Annemette; Kautz, Karl; and Nielsen, Peter A., "Making Sense of Project Management - A Case of Knowledge Management in Software Development" (2007). *ECIS 2007 Proceedings*. 86.
<http://aisel.aisnet.org/ecis2007/86>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2007 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

MAKING SENSE OF PROJECT MANAGEMENT – A CASE OF KNOWLEDGE MANAGEMENT IN SOFTWARE DEVELOPMENT

Annemette Kjærgaard, Department of Informatics, Copenhagen Business School, Howitzvej 60, 2000 Frederiksberg, Denmark, amk.inf@cbs.dk

Karl Kautz, Department of Informatics, Copenhagen Business School, Howitzvej 60, 2000 Frederiksberg, Denmark, karl.kautz@cbs.dk

Peter Axel Nielsen, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7E, 9220 Aalborg, Denmark, pan@cs.aau.dk

Abstract

How can a software company make sense of project management when it becomes involved in software process improvement? In software development most research has an instrumental view of knowledge management thus neglecting what is probably the most important part of knowledge management namely making sense of practice by developers and project managers. Through an action case, we study the knowledge management processes in a Danish software company. We analyse the case through the lens of a theoretical framework. The theoretical framework focuses in particular on sensemaking, collective construed realities and individual cognitive processes. It directs attention towards the creation and negotiation of construed realities and how collisions trigger moving from creating to negotiating. We discuss the value and relevance of this framework for software development companies and conclude that it provides substantial insight which could not have been achieved through an instrumental perspective on knowledge management.

Keywords: Knowledge management, software development, project management, sensemaking, explicit knowledge, tacit knowledge.

1 INTRODUCTION

Managing knowledge in software development has been of interest to IS researchers for quite a while. Examples are the experience factory (Basili & Green, 1994; Rus & Lindvall, 2002), the application of knowledge management theory to explain software process improvement (Arent & Nørbjerg, 2000; Baskerville & Pries-Heje, 1999; Kjærgaard & Kautz, Forthcoming; Mathiassen & Pourkomeylian, 2001), and the role of knowledge in software firms (Agarwal & Prasad, 2000; Althoff et al., 2000). Nevertheless the discussion on how to manage knowledge in software development has far from finished and new proposals as well as lessons learned are continually suggested. It remains a problem, though, that published research in the information systems field in general is largely grounded in a view that considers knowledge as an objective commodity, which can be collected, represented symbolically and processed like information (Dahlbom & Mathiassen, 1993; Tsoukas, 1998). As a consequence the literature shows a certain preoccupation with information technology and technical solutions while it reflects a limited view of individual and organizational knowledge related processes (Swan et al., 1999). The practice of knowledge management is frequently reduced to the implementation of new IT based systems, procedures for documenting and sharing information and the documents themselves. By focusing on externalisation and documentation of knowledge, important organizational aspects, in particular human and social issues, are overlooked.

This paper takes these issues into account and is based on a perspective of knowledge and knowledge management as an ongoing sensemaking process (Weick, 1995). Our focus is on the process of creating a shared handbook for project management in software development. By using sensemaking as the analytical lens, we explore how software developers make sense of the process of managing projects, and how this influences their perception of what should be included in a shared handbook.

Based upon an empirical study of this knowledge management process in a Danish software development company, we provide insight of how developers make sense of project management based on their construed realities (Isabella, 1990). We discuss the value and relevance of taking a sensemaking perspective and suggest further use of thereof. The question which guided our research can be summarized as: What general understandings of project management do the participants in the improvement project draw from when they contribute to the handbook, and how do they make sense of its content and use?

The paper is structured as follows. The next section presents our research approach and methodology. The theoretical framework, which we use to analyse our empirical findings, is described in section three, and the case company is introduced in section four. Section five holds the analysis of our findings, which are discussed in section six, where we also make suggestions for future research. Finally we provide our conclusions in section seven.

2 RESEARCH APPROACH AND METHODOLOGY

The research presented in this paper is interpretive. It is based on an empirical case study and action research in Rovsing Space, a Danish software company, where the development of a project handbook was started as part of a larger project focused on software process improvement. The project handbook development project was conducted in order to improve Rovsing's project management practices and to support knowledge sharing amongst the project managers. The project ran for more than 12 months and a first version of the handbook was presented after eight months. Our research comprises this time period, but also includes a larger contextual study and other improvement projects.

The roles and length of stay in the field have varied for the three authors of this paper. One author has been involved in the project as an action researcher throughout a three-year period (McKay & Marshall, 2001; Iversen et al., 2004; Mathiassen, 2002). This author was involved in development of

the handbook from the very early stages of the project, although he did not contribute to the actual writing process. A second researcher participated as an action case (Braa & Vidgen, 1999) or involved researcher (Walsham, 1995) for three months, contributing primarily to the interpretation of the case as sensemaking. A third researcher acted as an outside observer of the case study (Walsham, 1995) and primarily contributed in the analysis of the data. The combination of intervention, interpretation and collaboration between the three researchers with different levels of involvement was chosen to bring interpretive rigour to the project and to introduce new analytical perspectives on the case (Madsen et al., 2006). This research design counters the criticism of action research, which suggests it may easily reduce to mere consultancy (Baskerville & Wood-Harper, 1996).

The project handbook development process was part of a research collaboration project called Software Process and Knowledge. It was undertaken by several project managers appointed by the CEO, the CEO himself and three action researchers. Meetings were held at regular intervals, typically once a month. During these meetings the action researchers captured data by participating and observing while taking notes. Furthermore, a variety of documents such as the original project proposal, drafts of the handbook, email correspondence, minutes of meetings, company documents as well as project reports and deliverables were collected.

In line with the interpretive approach, the analysis of the project handbook development process has come about through an iterative process of interpretation and comparison of the data based on the use of the analytical lens of sensemaking (Weick, 1995) which forms the basis for the theoretical framework in (Kjærgaard & Kautz, Forthcoming) to be presented in the following section.

We explicitly sought to map the sensemaking process of the case onto the model. Every discrepancy between the case and the model in this initial mapping was discussed at length and in such detail that it could be resolved. For every instance where a significant interpretation of the case did not immediately fit well into the framework, great care was taken not to force the case and our interpretation into the framework.

3 THEORETICAL FRAMEWORK – A PROCESS MODEL OF KNOWLEDGE MANAGEMENT AS ORGANISATIONAL SENSEMAKING

To analyse the data, we use a process model which was developed in an earlier process study of establishing organizational knowledge management (Kjærgaard & Kautz, Forthcoming). The model was constructed to understand knowledge management as an organizational sensemaking process unfolding in the interplay between collective understandings of the organisation and individual members' meaning construction. In this study we use the model to explore the sensemaking processes which were in play in the process of constructing the project management handbook at Rovsing Space.

The model explains knowledge management as a sensemaking process unfolding in the interaction between collective and individual cognition.

The model operates with two sub-processes of the knowledge management process, *creating* and *negotiating*, each of which has a dominant *construed reality* as well as a set of *cognitive processes*. Furthermore, a triggering event of *collision* between expectations and experiences marks transition from one process to the other. Although there is no specific indicator for time, the change from one stage to the other indicates that the process unfolds over time.

The process model is shown in Figure 1.

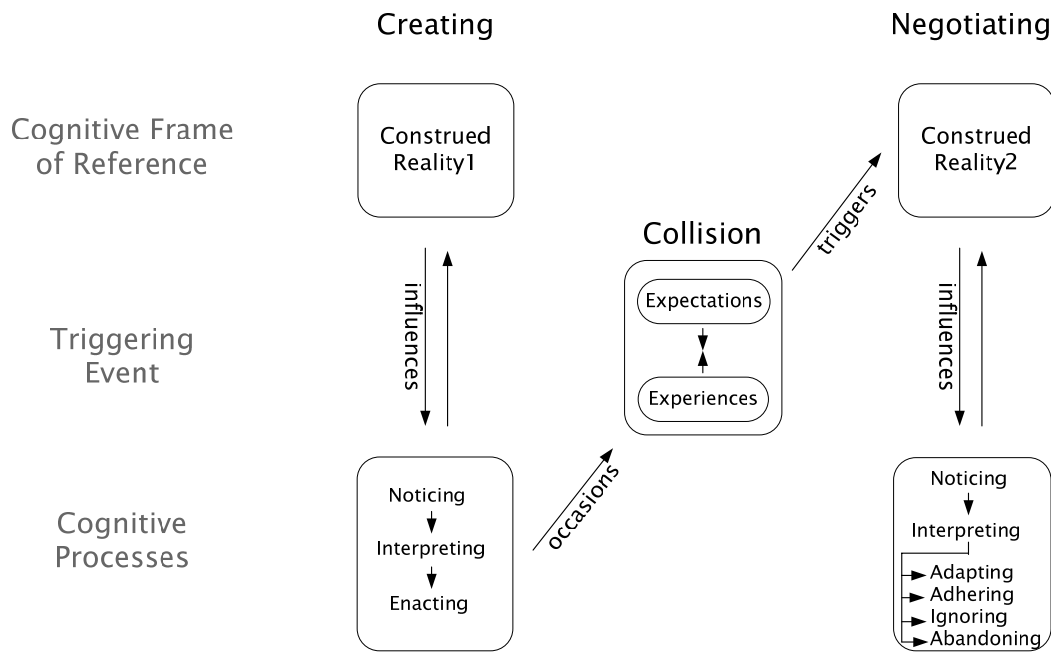


Figure 1 Basic flow of the process model of knowledge management as processes of sensemaking, based on (Kjærgaard & Kautz, Forthcoming)

The top layer of the model shows the organisational members' dominant frame of reference at the two stages of the process: creating and negotiating. These construed realities (Isabella, 1990) are collectively held perceptions of how to behave in the organization. A construed reality is an assembly of more or less connected pieces of information and beliefs, which together form a picture that confirms or constructs a reality (Isabella, 1990). It can be seen as a frame, and thus a substance of the sensemaking process into which the members' experiences as sensemaking cues can be placed (Weick, 1995). In the model only one construed reality is shown as the cognitive frame in each of the sub-processes of creating and negotiating. However, more construed realities can co-exist (Pratt & Foreman, 2000; Smircich & Stubbart, 1985). What we refer to in the model is therefore the dominant construed reality in each sub-process.

The bottom layer of the model shows the cognitive processes at the individual level, i.e., the cognitive processes which form the organisational members' sensemaking and meaning construction. The arrows from the construed realities to the cognitive processes show that the individual processes are influenced by the collectively held understandings of the organization.

The cognitive processes in the first period of creating are based on the understanding that members notice particular parts of the ongoing flow of information which they are exposed to continuously. What they bracket out of this flow is based on the dominant construed reality which guides their *noticing* (Weick, 1995). Following, they *interpret* the noticed information drawing from the construed reality, which they finally *enact*. The arrows from the cognitive processes pointing back to the construed reality shows that the process of enactment confirms and strengthens the construed reality. This cyclic process can be seen as a process of thinking in circles or a self-fulfilling prophecy (Weick, 1995; Weick, 1979).

Intermediating the two sub-processes of the knowledge management process and the two layers of cognitive processes, is a collision between expectations and experiences. It is the members' perception of dissonance between expectations and experiences which creates uncertainty and trigger the members to pursue stability by establishing a new frame into which their experiences fit. In sensemaking terms this collision between expectations and experiences can be referred to as an

occasion for sensemaking or a shock (Weick, 1995) which triggers a more intense process of meaning construction in order for the members to reinterpret the dominant construed reality or to create a new.

The cognitive processes in the second period of negotiating are more ambiguous than those of the first period as they are influenced by a new construed reality which dominates but has not necessarily replaced the construed reality of the creating process. Whereas sensemaking in the creating process is driven by beliefs in the form of expecting and arguing, sensemaking in the negotiating process is based on action as well as beliefs. The action which guides sensemaking in the negotiation process is primarily of the members' own creation (sensemaking as committing). By creating the initiatives on the basis of the first dominant construed reality in the creating process, the members commit themselves to this action. In the negotiating process this commitment brings the members to search for explanations to make sense of their own actions.

Depending on which of the construed realities that dominate their sensemaking process, the members create meaning out of their experiences and subsequently act upon this meaning. The cognitive processes in the process of negotiating thus reflect the extent to which the actors *adapt* their beliefs to the new construed reality and consequently adjust their activities to better fit the new situation. Alternatively, they *adhere* to the construed reality dominating the creating process and choose to leave the organisation to pursue their interests elsewhere, or they *ignore* their experiences and simply try again. Or finally, they accept the new reality and *abandon* the knowledge management action, acknowledging that it did not fit the new construed reality.

The empirical findings from our study will be analysed using the framework of knowledge management as sensemaking.

4 CASE DESCRIPTION – ROVSING SPACE

We did our empirical studies in the company, Rovsing Space, which develops software as a subcontractor to the European Space Agency (ESA). Rovsing develops a wide range of dedicated software for on-board, micro-gravity, verification & validation, ground station control, and check-out systems.

The company, founded in 1992, is a rather old company in the software development business, and it has lived through many changes. Most of the company's software developers have a M.Sc. in engineering or computer science. A large part of the software developers have developed software for the space industry for many years and are quite experienced with the particulars of space products. A perhaps equally large part of the software developers have little experience within the space industry, but rely on experience from other domains of software development.

In 2002 Rovsing decided to focus on improving its software processes and entered into a collaborative research project called Software Processes and Knowledge. The initiative to improve software processes started with a traditional, though light-weight, assessment of the current software processes, based on the understanding of the action researchers that improvement should be initiated with an assessment (Mathiassen et al., 2002) though not necessarily with a formal and model-based assessment (Iversen et al., 1999). Rovsing's current software processes were compared informally to the processes in the Bootstrap model (Kuvaja, 1999; Kuvaja et al., 1994). The result was significant discrepancies between the Bootstrap model and the company's current software processes and practices. This led to a decision to prioritize improvement of requirements engineering and on project management though other processes were also in need of improvement. The focus of this paper is on how Rovsing addressed improvement of project management.

The European Space Agency has a large number of standards that its subcontractors including Rovsing Space must comply with. A number of these are process standards and Rovsing has in the past dealt with the issues of compliance uniquely in each development project. The ESA standards are quite complex. They form a hierarchy of standards and the levels of details in instructions vary considerably

and so do whether instructions are required, recommended, or optional. Thus, the documentation of compliance is never trivial and it requires project managers to be well-read in the many ESA standards. This led to the first idea for improvement, namely that a new and improved project management process should be documented in a *handbook*. The handbook should be compliant with the relevant ESA standards and it was expected that it would be much easier to document the compliance when project management was performed in accordance with the guidelines in the handbook.

It is Roving's declared strategy to deliver fixed-priced software on time. The CEO in particular expressed a deep concern for achieving this goal and was very clear in maintaining that all improvement activities should be directed at this. There was a strong belief among some of the experienced project managers and the CEO that Roving already applied a number of *best practices*, i.e., established practices within the discipline and within the industry. This belief was also enforced by Roving already being in compliance with the ESA standards as these standards would then inherently reflect the best practices within the software space industry. This led to the idea that the software processes for project management should encompass these already existing practices. The idea seemed to be that they already knew what to do; it was just not documented in a shared handbook.

Other project managers were not as optimistic about the existing practices and wondered whether they were in fact industry-best practices which just needed to be enacted by project managers or whether they had to be improved. Their view was that not all the project managers had had the necessary training in these processes; they did not necessarily have the competence needed, and therefore some *training and education* would be necessary at least for new project managers. This led to the idea that a project manager education should be established based on the new and improved processes.

The improvement project started with a series of workshops with the purpose of: (1) designing and writing a handbook for project management and (2) designing an education or training course in the contents and use of the handbook. In the workshops, three to five experienced project managers participated together with the CEO and two to three action researchers.

During the first workshop, the (sub)processes of project management were identified. This identification was to a large extent based on the project managers' experience and only to some extent came from the ESA standards. Some of these processes were easily written by one to two project managers and they never led to any controversy in later workshops. A few significant processes were very difficult for the project managers to write. Although it was a design goal for the handbook to be brief, they eventually drafted quite complex process descriptions, which were then heavily criticised by other project managers at later workshops. This led to much iteration over the process descriptions which were expanded and condensed several times while the core ideas of the processes were constantly negotiated during the workshops.

The handbook was designed, written and reviewed through these workshops, but the aims of the workshops changed during the progress of the project and attention was diverted to related problems as well. The education or training course was never designed, and the handbook gradually expanded its scope to become a handbook for software engineering processes as well. The workshops were never chaotic, but there was disagreement about various issues. Most disagreements were overcome and problems were solved. However, a fundamental issue arose and gradually caught the attention of the participants during the last workshops. This issue was a concern for where the project managers' knowledge came from and how they could share this knowledge.

This issue arose from the participants' experience that there were two groups of processes: (a) the 'easy processes' which were easy because the project managers and the CEO shared the knowledge necessary to perform and describe the processes, and (b) the 'difficult processes' which were difficult because the participants did not share the necessary knowledge for writing the procedures. A few of the project managers and the CEO in particular were of the opinion that the handbook should be written and formalised and that all the project managers should then perform the formalised processes based on the instructions in the handbook. Several other project managers had the view that the

handbook should contain more than a set of instructions; it should also contain explanations that would enable a project manager to perform the processes.

To analyse these differences in more detail, we use a framework of knowledge management as sensemaking.

5 ANALYSIS

The theoretical framework, presented in section three, enables us to analyse our data from a knowledge management perspective, focusing on how participants made sense of the handbook as a means for process improvement.

In the first period of creating the handbook, two different construed realities were at play. First there was the construed reality of the CEO in which he sees the project management processes as mere descriptions that project managers are to follow rigorously. This construed reality became apparent during the workshops when we heard the CEO speak of why he wanted a handbook developed in the first place. Earlier in his career, the CEO was an experienced project manager and based on his own experiences, he challenged the project managers to manage projects in a much stricter way than they had previously done. His argument was that “if he could do it, they can as well”. The CEO’s construed reality bracketed his view of project management in Rovsing and guided what he noticed and interpreted and subsequently enacted. As a consequence, his contribution to the handbook reflected an emphasis on what plans and procedures should be followed in the project management process.

Secondly, the project managers generally shared a construed reality deeply rooted in their day-to-day experience with project management. Included in this experience was past projects which had been late, changing requirements negotiated with ESA managers, and plain difficulties of delivering products at the agreed time with the agreed features. To these project managers it did not make sense to simply stick to plans and follow procedures. In cases of problems in projects they rarely experienced that the problems could be alleviated by adhering to the plan or by re-planning. To them the problems were usually of the kind, where they had not anticipated these problems to arise, they had unknowingly allowed the ESA manager to introduce the trouble or a technical issue had proved to be a much bigger risk than planned late in the project. These project managers believed in planning, but only to a limited extent. In their noticing and interpreting of events and problems, they relied therefore only to a limited extent on plans and procedures, because a much stricter process did not make sense to them. What they enacted, when constructing the handbook, was therefore a much more flexible view of project management as a constantly changing process, which could not be controlled by plans and procedures.

During the first workshops both of these construed realities influenced the discussions and the development of the project management handbook. The different ways of interpreting and enacting the contents of the handbook was based on different construed realities which led to much confusion in discussions that could not immediately be reconciled. Gradually, it became apparent that in the common wish to develop the handbook, participants experienced a *collision* between how the project managers and the CEO respectively perceived project management in software development. The CEO expected the project managers to be able to avoid project overruns and maintain the initially negotiated budget and thus contribute to profitability of the company as a whole. The CEO and the project managers shared the experience that there had been too many incidents in the past where these expectations had not been fulfilled. However, the project managers found the CEO’s expectations less realistic and thought that he neglected the kind of trouble, they faced.

Only gradually did this collision lead the workshop participants to realize that the views embedded in the project management handbook needed to be negotiated. During the *negotiation process* the CEO and the project managers came to a larger degree to share a construed reality. In this new shared construed reality there was a sense that the project managers already had the competence to perform most of a project manager’s tasks. However it was also acknowledged that sometimes the CEO had to

insist that contracts and agreements were adhered to and that plans were agreements between a project manager and the CEO. Moreover, an understanding emerged that project managers should use all possible means (e.g., written documentation, plans, checklists, procedures) to remember all important and relevant issues because forgetfulness and lack of anticipation were bound to lead to troubled projects. This negotiation of 'reality' is illustrated by three examples in the PM handbook: a technique for estimation, a template for a project plan, and a procedure for shipment.

5.1 A technique for estimation

A huge debate arose when the first draft for how to conduct estimation in projects was reviewed. The draft included several important concepts of estimation, gave a number of techniques and ended by promoting 3-point estimation, which was then explained in some detail. The draft was written with the intention of explaining estimation techniques also for those that had not previously seen nor applied estimation techniques. The most common way to estimate among project managers at the time was guessing based on the project manager's past experience, but there was significant variation in how the project managers estimated projects and tasks. The draft was briefer than a standard textbook on software engineering, but it covered similar contents. During the discussion there was a strong feeling that these techniques would not improve the estimation accuracy, but also that the draft was either too detailed for those who already knew the techniques or too abstract for those who did not. After a long discussion the author of that draft section was persuaded to rewrite it.

At a later workshop the second draft was discussed. That draft contained only the recommended 3-point estimation and a brief terminology on different estimates used by both ESA and Roving. It was evident that for new project managers the PM handbook could not substitute training and education in estimation, but it did provide the necessary common ground and it was compliant with ESA standards.

At the final approval of this section of the handbook the construed reality had reached a state in the project managers' perception where they were very likely to adhere to this part of the process.

5.2 A template for a project plan

The variety of project plans was significant. ESA did not require a particular way, nor did Roving internally. Most of the difference between project plans could not be explained by the difference between the projects' conditions and contents. There was a genuine desire among the project managers to formulate a template that would remind the project managers of what was important in a project plan. This was strongly supported by the CEO. The CEO particularly wanted the project plan to be viewed as a contract between him and the project manager. They should go through the details in the project plan before it was approved and it should be monitored and discussed at regular meetings between the CEO and the project manager. It was important that there was agreement on what the contents of a project plan should be. After this was clarified during the second workshop, it was relatively easy for the author of this section of the handbook to rewrite it, and it was later approved.

The template in its final form would stipulate headings and contents in the project plan, but it would also stipulate which states the plan should go through (e.g., draft, approved) and how the project manager and the CEO would discuss monitoring of progress. By the end of the last workshop there was agreement among the project managers that the template was highly relevant and that they would begin using it. It was very general and in every instance, where it was to be used, it had to be adapted to suit the specific needs of the project.

5.3 A procedure for shipment

There had been several incidents in the past where projects had not been able to deliver products to the customer on time or with items missing due to troubles with packaging and shipment. A procedure was suggested by the CEO that would ensure, in detail, that all possible problems in the final shipping

were taken care of in due time. The procedure included very detailed instructions, e.g., on shipping of hardware in wooden boxes if necessary, on customs clearance and on assembly manuals. At first the project managers found it much too detailed. It was in sharp contrast to other parts of the handbook where many details had been left out. After much discussion it was evident that the procedure was really to be viewed as a checklist. A checklist had the advantage of reminding a very busy or otherwise forgetful project manager of small, but crucial issues to deal with in due time.

The difference between the procedure for shipment and the technique for estimation is that shipment is a routine task, but it can be destroyed by forgetting just a small part. Estimation on the other hand is not routine and appropriate estimation relies on the project managers' competence and experience. It cannot become routine to the same degree. After the realization that the handbook could easily contain several types of processes – some highly dependent on competence and some highly routinized – the project managers agreed to adhere to the procedure.

6 DISCUSSION

The handbook consists of elements varying from procedures to be followed to general guidelines that require the project managers to possess considerable knowledge and competence in advance.

Our case shows that the immediate impression that the project management handbook should be seen as an externalisation of the project managers' knowledge about project management is too simplistic. While the project managers' knowledge does indeed contribute to the handbook, this knowledge is far from unequivocal and in some cases it clashed with the knowledge of the CEO. The case shows that the participants in the project drew from different construed realities of what project management should be and had been in the past. These different construed realities bracketed what they noticed to be central or important about project management and thus what they suggested should be included in the handbook. In other words, they did not see what the others saw as important about project management, and it is doubtful whether the handbook would have been of any use to them, if they had not been through a process of collective sensemaking to create a shared construed reality, giving sense to the content of the handbook (Gioia & Chittipeddi, 1991). The findings from the case suggest that the sensemaking process is a vital part of the process of constructing a handbook. By analysing how the participants' sensemaking processes unfold, a more complex understanding is created of how the participants perceive the handbook and what they accordingly suggest it should contain.

Another related issue, which became apparent in the study, was the consequence of a predominant rational understanding of project management combined with a likewise predominant rational understanding of knowledge management. Project management and also software process improvement in software engineering is often presented through a very instrumental view on knowledge management. Knowledge management is taken to be a question of what should be stored in a database and how the data in the base should be searched (Althoff et al., 2000; Conradi & Dingsoyr, 2000; Lindvall et al., 2001). There has already been some critique voiced on this view based from a theoretical stance, e.g., (Aaen et al., 1998). Our case study shows from an empirical stance that the handbook as a storage of knowledge is very limited and misses the significance of the project managers' experiences and competences. The same can be said about knowledge management where a technical-rational understanding has also dominated (Swan et al., 1999). The consequence of this was also apparent in our study where the rational approach to project management, primarily promoted by the CEO, led to an overly rational approach to knowledge management. Knowledge management was initially limited to focus on the process of externalising the participants' knowledge into written procedures and thus brought about a quite simplistic view of what forms of knowledge are important in project management. It also led to a quite ineffective knowledge creation process. However, triggered by the collision of this approach with the project managers' experiences, the negotiation process brought to light a more complex reality. This reality was subsequently externalised to a certain extent, but as explained by Tsoukas (1996) acknowledging that explicit and tacit knowledge is

intrinsically interrelated and mutually constituted, and that all knowledge thus has a tacit dimension, it more importantly became part of the group's tacit knowledge.

Also some implications for practice can be drawn from the study. Our empirical findings point to the issue of who takes ownership of the handbook and its contents. Ideally, the CEO and the project managers together should take responsibility and this is often assumed to be the case in organizations which decide to create some form of knowledge repository. However, our analysis shows that ownership has to be negotiated between the contributors to the handbook in a process of collective sensemaking. In other words our findings indicate that the negotiation process is itself important for making clear what understanding of reality guides the contributors' contributions to the handbook. By focusing on the process and the ongoing negotiation, a much more complex understanding of what is knowledge emerges.

7 CONCLUSION

The case and the analysis illustrate a set of complex knowledge processes that the framework has helped us untangle. The types are briefly summarised as the creation and negotiation of knowledge. Where creation of knowledge initially relied on how the participants made sense of project management based on their own construed realities, the negotiation of knowledge invoked a renewed sensemaking process. In the negotiation process the participants reconsidered their construed realities and constructed a new shared construed reality based on social interaction, which helped them to make sense of the various approaches to project management, which were held among the participants of the group.

In conclusion, our findings show that by applying a sensemaking framework, we gain a better understanding of the complexity involved in creating a shared object for knowledge management in software development. Whereas the dominant view in the IS literature on knowledge management still focuses on the externalization of knowledge, the sensemaking framework enables us to understand what general understanding the participants draw from when they contribute to the creation of organizational knowledge. The sensemaking framework and the analysis of our case may have crucial effect on software development and software process improvement. The effect should be that an improvement effort would need to go through the two processes of creating and negotiating.

References

- Agarwal, R. and Prasad, J. (2000) A field study of the adoption of software process innovations by information systems professionals. *Ieee Transactions On Engineering Management* 47 (3), 295-308.
- Althoff, K.-D., Bomarius, F. and Tautz, C. (2000) Knowledge management for building learning software organizations. *Information Systems Frontiers* 2 (3/4), 349-367.
- Arent, J. and Nørbjerg, J. (2000) Software process improvement as organizational knowledge creation -a multiple case analysis-. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, p 11, Wailea, Hawaii.
- Basili, V. and Green, S. (1994) Software process evolution at the sel. *IEEE Software* 11 (4), 58-66.
- Baskerville, R. and Pries-Heje, J. (1999) Knowledge capability and maturity in software management. *The DATABASE for Advances in Information Systems* 30 (2),
- Baskerville, R. and Wood-Harper, A. T. (1996) A critical perspective on action research as a method for information systems research. *Journal of Information Technology* 11, 235-246.
- Braa, K. and Vidgen, R. (1999) Interpretation, intervention and reduction in the organizationl laboratory: A framework for in-context information systems research. *Accounting, Management and Information Technologies* 9, 25-47.

- Conradi, R. and Dingsoyr, T. (2000) Software experience bases: A consolidated evaluation and status report. In *Product focused software process improvement*, pp 391-406.
- Dahlbom, B. and Mathiassen, L. (1993) *Computers in context - the philosophy and practice of systems design*. Blackwell, Cambridge, UK.
- Gioia, D. A. and Chittipeddi, K. (1991) Sensemaking and sensegiving in strategic change initiation. *Strategic Management Journal* 12 (6), 443-448.
- Isabella, L. A. (1990) Evolving interpretations as a change unfolds: How managers construe key organisational events. *Academy of Management Journal* 33 (1), 7-41.
- Iversen, J., Mathiassen, L. and Nielsen, P. A. (2004) Managing risk in software process improvement: An action research approach. *MIS Quarterly* 28 (3), 395-433.
- Iversen, J., Nielsen, P. A. and Nørbjerg, J. (1999) Situated assessment of problems in software development. *The DATABASE for Advances in Information Systems* 30 (2), 66-81.
- Kjærgaard, A. L. and Kautz, K. (Forthcoming) A process model of establishing knowledge management: Insights from a longitudinal field study. *OMEGA*, available online 11 September 2006.
- Kuvaja, P. (1999) Bootstrap 3.0—a spice1 conformant software process assessment methodology. *Software Quality Journal*, (8), 7-19.
- Kuvaja, P., Similä, J., Krzanik, L., Bicego, A., Saukkonen, S. and Koch, G. (1994) *Software process assessment & improvement - the bootstrap approach*. Blackwell Publisher.
- Lindvall, M., Frey, M., Costa, P. and Tesoriero, R. (2001) Lessons learned about structuring and describing experience for three experience bases. In *Learning Software Organizations (LSO)* (Althoff, K.-D. and Feldmann, R. L. and Müller, W., Eds), pp 106-119, Springer Lecture Notes in Computer Science LNCS #2176.
- Madsen, S., Kautz, K. and Vidgen, R. (2006) A framework for understanding how a unique and local is development method emerges in practice. *European Journal of Information Systems* 15 (2), 225-238.
- Mathiassen, L. (2002) Collaborative practice research. *Information Technology & People* 15 (4), 321-345.
- Mathiassen, L. and Pourkomeylian, P. (2001) Knowledge management in a software process improvement unit.
- Mathiassen, L., Pries-Heje, J. and Ngwenyama, O. (Eds.) (2002) *Improving software organizations: From principles to practice*. Addison-Wesley, Reading, MA.
- Mckay, J. and Marshall, P. (2001) The dual imperatives of action research. *Information Technology & People* 14 (1), 46-59.
- Pratt, M., G. and Foreman, P. O. (2000) Classifying managerial responses to multiple organizational identities. *Academy of Management Review* 25 (1), 18-42.
- Rus, I. and Lindvall, M. (2002) Knowledge management in software engineering. *IEEE Software* 19 (3), 26-38.
- Smircich, L. and Stubbart, C. (1985) Strategic management in an enacted world. *Academy of Management Review* 10 (4), 724-736.
- Swan, J., Scarbrough, H. and Preston, J. (1999) KM - the next fad to forget people. In *Proceedings of the 7th European Conference on Information Systems* (Pries-Heje, J. and Ciborra, C. U. and Kautz, K. and Valor, J. and Christiaanse, E. and Avison, D. and Heje, C., Eds), Copenhagen, Denmark.
- Tsoukas, H. (1998) The word and the world: A critique of representationalism in management research. *International Journal of Public Administration* 21 (5), 781-817.
- Walsham, G. (1995) Interpretive case studies in is research: Nature and method. *European Journal of Information Systems* 4, 74-81.
- Weick, K. E. (1979) *The social psychology of organizing*. McGraw-Hill, New York.
- Weick, K. E. (1995) *Sensemaking in organizations*. Sage, Thousand Oaks, CA.
- Aaen, I., Bøttcher, P. and Mathiassen, L. (1998) The software factory: Contributions and illusions. In *6th European Conference on Information Systems (ECIS)* (Baets, W. R. J., Ed), pp 736-750, Aix en Provence, France.