A Balanced Theory of Knowledge Management in Software Process Improvement

Karlheinz Kautz
School of Business IT and Logistics
RMIT University
Melbourne, Australia
Email: karlheinz.kautz@rmit.edu.au

Gro Bjerknes
Information Technology
Monash University
Melbourne, Australia
Email: gro.bjerknes@monash.edu

Bo Hansen Hansen
StrayCorp
Copenhagen, Denmark
Email: bo@straycorp.dk

Abstract

Knowledge management (KM) plays a prominent role in IT and software development companies. It is often introduced as part of larger organisational change processes that aim at improving their software development processes. Process improvement in the IT sector has led to the establishment of the software process improvement (SPI) discipline. Based on a study of the KM and SPI literature we offer a framework for how software companies can ground their improvement activities founded on an alignment of KM and SPI. We identify two archetypes of knowledge organisations which we label exemplary and situational and two approaches to process improvement, which we call normative and reflective. Our analysis of the relationship between KM and SPI leads to a proposal for a balanced theory of KM in SPI and provides valuable insights into how meaningful KM can be conducted for process improvements in IT and software organisations.

Keywords: knowledge management, software process improvement, theory development.

1 Introduction

In the digital age the primary asset of post-industrial organisations is no longer the physical equipment and production environment, but rather the know-how of the employed work force (Quinn *et al.*, 1996). Digital innovations, products and services are based on information technology which creates new opportunities for low marginal costs, low distribution costs, and global reach (Shapiro & Varian, 1999). This strengthens the competition and introduces rapid changes to the organisation's environment. At the same time the customers are growing in sophistication and are increasing their demands (Davenport & Prusak, 1998; Bjerknes & Mathiassen, 2000). The key asset for companies thus is their ability to develop and utilise the intellectual competences of their employees to create services of value for their customers (Quinn *et al.*, 1996). This shift towards a knowledge society presents a change of environment that every organisation must handle (Liebowitz & Beckman, 1998) and has led to the establishing of the knowledge management (KM) discipline (Swan *et al.*, 1999).

KM plays a prominent role in knowledge intensive organisations. A major challenge in the KM field is how to facilitate identification, creation, and sharing of valuable knowledge in an organisation. In this respect IT and software development companies are no exception, and they too, in their pursuit of greater professionalism, continuously have to improve their performance. The development of information systems and digital innovations is dependent upon knowledge of the application domain and development practices, which is why the knowledge management field attracts more and more attention in the IT and software development community, both academically and in industry (Dingsøyr, 2001). In IT organisations KM is introduced, often as part of larger organisational change processes that aim at improving their software development process (Kautz & Nielsen, 2004). The concern for process improvement in the IT sector has already more than 25 years ago led to the establishment of, what was formerly called, the software process improvement (SPI) discipline and to research in this area (Humphrey, 1989).

The work we report here was part of a project that explored how knowledge management and software process improvement can support the formation of a learning IT and software development organisation (Nielsen & Kautz, 2008; Hansen, 2009). In this paper we present the conceptual basis for investigating how software companies, on an organisational level, can ground and strengthen their improvement activities through the establishment of a learning software organisation based on concepts from knowledge management. To this end this paper's objective is to answer the following research question: *What is the conceptual relationship between KM and SPI?* The research question is answered through a study of seminal knowledge management texts presented in the next section and the software process improvement literature presented in section 3. We identified two archetypes of knowledge organisations which we label exemplary and situational knowledge organisations and found two approaches to process improvement, which we call normative and reflective approaches to software process improvement. On this background the relationship between KM and SPI is discussed in section 4 and leads to a proposal for a balanced theory of KM and SPI in section 5. Our research is summed up with some conclusions in section 6. The study thus contributes both to the knowledge management and the process improvement disciplines with valuable insights into how meaningful knowledge management can be conducted for process improvements in IT and software organisations.

2 Characteristics of Knowledge Management and Knowledge Organisation Archetypes

In line with Quinitas *et al.*(1997) we understand KM as any process or practice of creating, acquiring, capturing, sharing and using knowledge wherever it resides to enhance learning and performance in organisations including the creation of environments in which learning and knowledge exchange can take place. We identify in the KM literature two archetypal organisational forms with regard to knowledge management which we label the *exemplary* and the *situational* knowledge organisation. The exemplary archetype is based on the notion that knowledge is, and can be explicated, whereas the situational knowledge archetype is based on the notion of tacit knowledge. The archetypes are distinguished by their fundamental differences concerning the following characteristics: *1*) knowledge type, 2) knowledge creation approach, *3*) learning type, *4*) knowledge retrieval type, and *5*) knowledge management strategy.

The primary knowledge type in an organisation might be explicit or tacit (Polanyi, 1966). The organisation's capabilities to recognise one or the other of these knowledge types affects the organisational settings and thus the prevailing organisational routines for acquiring and sharing knowledge. If the primary knowledge type in an organisation is explicit the organisation's knowledge management and knowledge sharing routines will focus on explicating the organisation's member's knowledge. Often this will involve strong document recording, classification, searching, and distributing capabilities (Hansen *et al.*, 1999). On the other hand, if the prevailing knowledge type is the tacit form, the organisational routines established to secure knowledge transfer and diffusion will largely be executed by the staff of the organisation. The primary routines will target the abilities for employees to locate each other, and each other's expertise, and further the routines will provide means for close collaboration and education (Hansen *et al.*, 1999). The knowledge creation approach is either based on knowledge exploitation or on knowledge exploration (March, 1991; Levinthal & March, 1993). In an organisation which bases its knowledge creation on knowledge exploitation the ability to analyse and fully understand the already established practices is important. Fine tuning these and continuous optimising of the known settings will constitute process development. Thus, the parts of the organisational routines that are concerned with knowledge creation will be established by expert systems in which a detailed model of the company's business area will be maintained. Feedback routines will secure that experiences from practise will further optimise this model. In an organisation which bases itself upon knowledge exploration the purpose of the knowledge routines will be to challenge the established model of the business area in question, and thus to seek fundamentally new approaches or meanings within the business domain. Such routines will be dependant of the ability to cross examine and to interrelate knowledge of any type, and to focus on establishing the right personal relations between experts in various fields.

The prevailing learning type is either single loop learning or double loop learning (Argyris & Schön, 1974). Organisations practising single loop learning are striving for expertise based on fast and robust feedback mechanisms. The strengths of the single loop paradigm is the quick way of reviving experiences and utilise them to gain more knowledge concerning well defined and scoped areas of interest. This way it is possible to make relatively quick decisions based on experiences and complete learning cycles (Hedberg, 1981). Organisational decisions based on rule systems and detailed process descriptions fit well with this learning type. Double loop learning involves more thorough analyses and requires the questioning of the underlying assumptions upon which the existing knowledge has been build. Organisations in which this is demanded must secure these skills e.g. by hiring highly educated experts and by establishing routines that facilitate innovation (Christensen *et al.*, 2002; Charitou & Markides, 2003). The knowledge retrieval type applied in an organisation can also have two forms: It can be automatic or controlled (Walsh & Ungson, 1991). The automatic knowledge retrieval is characterised by relying on well-established or habitual sequences of action. In an organisation relying on this type of knowledge retrieval standard operating procedures and heuristics can facilitate a quick and effortless decision making when decisions are needed in situations where the context is known and thus predictable. The organisation's decision making apparatus will have pre-interpreted earlier experiences and will have condensed these into routines and guidelines. The task thus is to maintain the model and constantly adjust it. The controlled knowledge retrieval is characterised by thorough analysis of situations which are more complex and in which the parameters are unknown to the decision apparatus, and therefore need more analytic effort to be understood. In this way fundamentally new explanation models are constructed to fulfil new requirements and provide new and specialised answers. In this situation the organisational settings will be optimised towards the ability to conduct such deep analyses. This could be by facilitating cooperation e.g. in expert teams consisting of experts from different knowledge areas or with different skills.

Finally, the preferred knowledge management strategy can either be a codification or a personalisation strategy (Hansen *et al.*, 1999). The codification strategy relies on knowledge being codified in various forms to be easily shared among employees in the organisation. This suggests that the organisational routines for storing and retrieving data concentrate on procedures for explicating and characterising the experiences, e.g. into databases or document repositories. The personalisation strategy on the other hand relies on people as bearers of knowledge and at the same time people as the pivot around which the organisation shares its knowledge. In such an organisation the routines should facilitate easy location of experts and their knowledge as well as collaborative means for sharing this knowledge. Using these five characteristics we can now describe the exemplary and the situational knowledge organisation. In practise the five elements are not as clear cut and are intertwined in a complex way. They thus position a specific organisation somewhere in between the two archetypes (see figure 1).

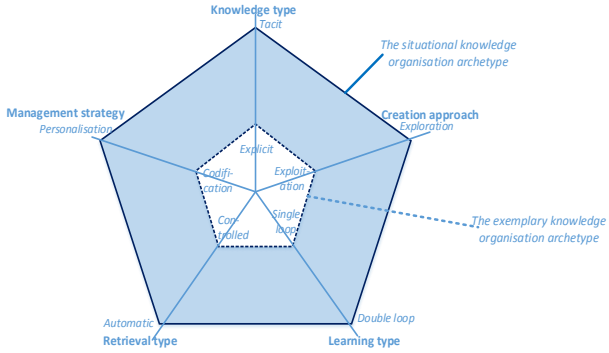

Figure 1: Knowledge management characteristics and their relation to knowledge organisation archetypes

2.1 Exemplary Knowledge Organisations

The exemplary organisation is characterised by conforming to an ultimate form of perfection. In this organisation the primary knowledge asset is explicit knowledge (Polanyi, 1966). This knowledge is acquired through continuous knowledge exploitation (March, 1991; Levinthal & March, 1993) through which the organisation refines its knowledge concerning its business domain to a level where automatic retrieval (Walsh & Ungson, 1991) can be exercised in an efficient way. All relevant processes are known and given a specific scenario the organisation can prescribe a best practice to achieve its goals. Its complex rule-set is maintained by optimising the underlying model via single loop learning (Argyris & Schön, 1974) and the organisation is practising a codification strategy to share knowledge (Hansen *et al.*, 1999). In this type of organisation the focus is on understanding the business domain to an extent that every parameter is known and described including how it correlates to every other relevant parameter in the domain. The model of the business domain becomes the centre of the business. Creating business processes which support this model is the crucial task of the organisation. This type of organisation is very efficient as long as the model is correct and as long as the domain stays unchanged or only changes in small increments (Hansen *et al.*, 1999).

2.2 Situational Knowledge Organisations

The other archetype is the situational knowledge organisation. Situational here refers to the organisation acknowledging its position in relation to its surroundings. This means that in a situational knowledge organisation these surroundings act on an individual or organisational level to condition behavioural patterns. Therefore in this type of organisation the specific context in a given scenario is also the key element to the organisational acting. The primary asset is the ability to understand any given situation by utilising the employees' capabilities and as such the tacit knowledge (Polanyi, 1966) is the organisation's key asset. This knowledge is acquired in the process of solving tasks in the business domain and therefore is anchored in the practice of the organisation; it is by definition not easily codifiable. The ability to explore (March, 1991; Levinthal & March, 1993) this knowledge in different scenarios is the primary means of transferring knowledge between members of the organisation. The preferred strategy is that of personalisation (Hansen *et al.*, 1999) and the processes of the organisation focus on matching the employees to the specific tasks and on facilitating the employees' ability to retrieve the organisational knowledge in a controlled way (Walsh & Ungson, 1991) which allows for and supports double loop learning (Argyris & Schön, 1974) by continuously challenging the underlying paradigm. This type of organisation is most efficient when the challenges are hard to categorise and require specialised solutions.

3 Approaches and Contributions to Software Process Improvement

Software Process Improvement is an applied discipline grounded in the software engineering and information systems disciplines. It deals primarily with the professional management of IT and software firms, and the improvement of their practice, displaying a managerial focus rather than dealing directly with the techniques that are used to develop software. In terms of its theoretical heritage, SPI is equally indebted to the software engineering tradition and the total quality management (TQM) movement (Deming, 1982). Based on Hansen et al.'s (2004) literature review we categorise approaches and contributions to SPI from the SPI literature along two dimensions. For the first dimension we use an analysis framework inspired by Mintzberg's work (1990) on strategy

formation where he differentiated prescriptive and descriptive schools. We distinguish prescriptive approaches and descriptive contributions, the latter being writings that describe experiences of improvement programs in IT and software organisations. On the other dimension – following SPI terminology as identified by Aaen *et al.* (2001) – we distinguish normative approaches which, based on a standard or model, prescribe how SPI should be performed and reflective contributions which are concerned with any theoretical grounding, analysis, theory building, or reflection. On this basis we identified normative prescriptive approaches, normative descriptive contributions, reflective descriptive contributions, however no reflective prescriptive approaches.

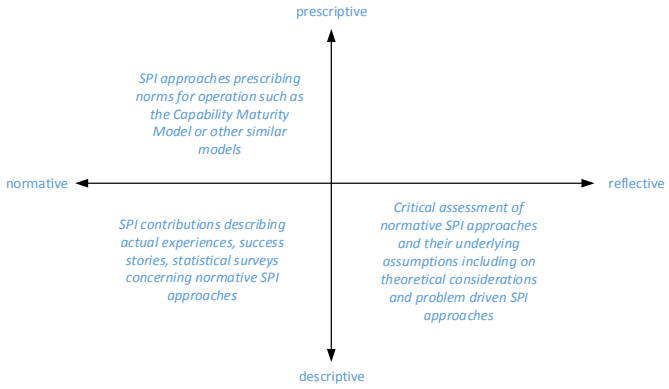

Figure 2: Approaches and contributions to SPI

3.1 Normative Prescriptive Approaches

Normative prescriptive approaches to SPI display a common set of characteristics. They focus on software development processes at the organisational, project, team, or individual level, and are concerned with measuring, standardising, and improving those processes. They prescribe norms for how individuals, teams or organisations should operate, and for how processes should be standardized and improved. They assume that processes can be measured, both as a baseline for improvement and to provide indications of subsequent improvements. They normally assume that well-understood software development processes exist that everyone agrees can be recommended in all situations. Organisational improvement is normally related to a maturity ideal: the mature organisation has articulated, standardised, measurable software development processes and measures them in order to learn how to improve them further. Maturity levels can be measured, using various questionnaire based techniques, and 'immature' organisations should normally follow a prescribed road-map to achieve the next maturity level. The Capability Maturity Model (CMM) is probably the best known and most widely used approach to SPI. CMM is formally defined as "a description of stages through which software organisations evolve as they define, implement, measure, control and improve their software process" (Paulk *et al.*, 1995). The CMM model describes how companies can mature according to specific stages. CMM describes five levels of maturity, against which a software organisation can be assessed. On level 1 – the initial level –ad hoc or non-deliberate actions define the orientation of the company. Level 2 – the managed level – is the stage where the organisation manages its processes and its projects according to these processes. Level 3 – the defined level – is where the processes are defined and interlinked into a coherent process framework. Monitoring of tasks is implemented which project management can rely upon during the daily planning and managerial tasks. On level 4 – the quantitatively managed level – the organisational processes are defined in detail, and the underlying cause-effect relations are known to a degree where quantitative monitoring measures the improvement, and points to areas for further improvements. On level 5 – the optimised level – an organisation engages in continuous learning and improvement. Other norm-based approaches are the BOOTSTRAP methodology (Kuvaja *et al.*, 1994) which combines elements of CMM with the relevant ISO, Defense and European Space Agency software standards, TAPISTRY, a software process improvement approach tailored for small enterprises (Kuvaja *et al.*, 1999) and SPICE (Software Process Improvement and Capability dEtermination), a standard for software process assessment (ISO, 2004). Another type of normative approaches focuses on solving organisation-specific problems. These problem-driven approaches to SPI (Aaen *et al.*, 1998) focus on ways to identify and solve specific problems in a software organisation based on an understanding that software development is a standardised, repeatable process which consists of sub-processes and procedures, which can be defined to a certain level of detail. This has led to the introduction of the concept of software factory of which the so called experience factory based on learning from documented experiences is a well-known example (Basili *et al.*, 1994).

3.2 Normative Descriptive Contributions

Normative descriptive contributions take as their principal focus the reporting of actual SPI initiatives in companies. Much of the normative descriptive work relates to experiences with the CMM. A distinct subcategory of these descriptive contributions – success stories – are reports of successful projects often written by people heavily involved in the projects, such as CMM consultants and SPI project managers. Examples are: Hughes Aircraft (Humphrey *et al.*, 1991), Motorola (Diaz & Sligo, 1997), NASA's Goddard Space Flight Centre (Basili & Caldiera, 1995). Success stories tend to present a generally positive tone about the SPI initiative described, and the narration of the success is combined with a presentation of problems encountered, lessons learnt, and advice for practitioners, which are, however, not generalised to theory. The problems described do not challenge the underlying paradigm, but relate to the operationalisation of the prescribed approach in the given context. A different approach to establish the benefit of SPI initiatives is found in a category that we label statistical surveys. These contribute to the SPI field by investigating very different subjects: CMM in small businesses (Bilotta & McGrew, 1998), the results and benefits of maturing (Johnson & Brodman, 1996), the difficulty of examining return on investment through CMM (Johnson & Brodman, 1996), success factors of CMM (El-Emam *et al.*, 2001). A number of more independent research oriented case studies of SPI initiatives exist as well, where the researchers use analytical frameworks, but also within the underlying paradigm. The introduction of a metrics program in a large Danish company has been reported in this way (Iversen & Mathiassen, 2003). Such research has also be done on software process improvement and organisational learning (Arent & Nørbjerg, 2000), on SPI in small companies (Kautz *et al.*, 2002), and on software process improvement through reflective practitioners (Börjesson & Mathiassen, 2003).

3.3 Reflective Descriptive Contributions

The reflective descriptive literature is sparse and differs in style and purpose. Topics of discussion range from the core assumptions of the CMM to the building of theoretical frameworks. Early contributions are focus on the CMM, while the later tend to have a broader view of the SPI field. Bach (1994) e.g. takes a critical look at the CMM and CMM assessments and argues that CMM has no formal theoretical basis and little empirical support, that it ignores people, reverses the institutionalisation of process for its own sake, and that it introduces an artificial goal – achieving a higher CMM level – instead of the goal of developing 'better' software. Other strands of this literature compare the CMM and other approaches (Lyytinen *et al.*, 1998) or analyse and discuss the CMM from a theoretical standpoint. Ngwenyama and Nielsen (2003), e.g., investigate the underlying values of the CMM, and reveal some contradictory assumptions made by the model about organisational culture. Saiedian and Chennupati (1999) provide an independent analytical framework for the evaluation of different software process models focusing on goals of the model, scope, domain, structure, management role, use of metrics, benefits, underlying models, rating process, and its organisational impact. Aaen *et al.* (2001) build a conceptual map to characterize the field of SPI using defining features such as management, approach and perspective of SPI. They introduce the distinction between model- driven and problem-driven SPI approaches which we to some extent applied in the discussion of normative approaches and contributions as a reflective contribution: Model-driven approaches are based on an underlying normative model of software process improvement which usually includes an explicit or implied normative model of software development — the processes to be improved. The main purpose for a SPI initiative is to align the software firm with this underlying model. Problem-driven approaches prescribe how a software organisation can improve its problem identification and solving activities, and thus become better at identifying which parts of the development process need to be improved, and how to address this task, and by this introduce double loop learning.

3.4 The Continuum of Software Process Improvement Approaches

The three identified categories of software process improvement approaches and contributions are distinguished based on whether they have a primarily normative prescriptive, normative descriptive, and reflective descriptive perspective. As the normative descriptive contributions overly report implementations and outcomes from the application of normative approaches they are not relevant for our further conceptual considerations. If the remaining two categories are viewed as a continuum instead of distinctive separate categories they form the end points that span from normative to reflective approaches. The characteristics of a specific SPI approach can then be used to position it in this range. The continuum represents an extension of the introduced distinction between model based and problem based normative SPI approaches (Aaen *et al.*, 2001) as introduced earlier. Instead of including only normative approaches in the distinction we broaden its scope to include all approaches that do not prescribe a specific norm for improvement, viz. those that are reflective in character. Thus

any improvement approach that relies strictly on prescriptions formulated in rules and norms would be placed on the normative end on the continuum. Similarly any approach that seeks fundamentally new solutions targeted directly to the specific context in an organisation would be placed towards the reflective end on the continuum. Most SPI approaches include a mix of the two approaches and thus will position themselves somewhere between the end points of the continuum. The same approach might be positioned differently depending on how it is used in a specific situation; e.g. the different stages of the CMM can be positioned differently even if they are derived from the same base approach. Although the CMM fundamentally is a normative approach introducing a specific set of rules, it can be argued that the higher levels 4 and 5 in the model, although still prescriptive in nature, include a more reflective approach. To achieve these levels, an organisation must be able to manage its knowledge in a proper manner, which is why software companies increasingly became interested in knowledge management (Dingsøyr, 2001; Kautz & Thaysen, 2001).

4 Software Process Improvement is Knowledge Management!

The objective of SPI is to provide improvement to the practice of software development companies. Software development companies are knowledge intensive companies and, as such, their practice relies on how well they manage their intellectual capabilities. Although many different SPI approaches exist in many different variations, and the scientific description of these approaches mainly is of a normative character, all SPI approaches presented here, to a certain extent recognise the importance of knowledge in modern organisations and therefore refer to collecting and utilising organisational knowledge. The focus on managing knowledge in software companies suggests that the SPI field has much in common with the KM field. This similarity in objectives is also recognised in several studies. Kautz and Nielsen (2004) describe how SPI innovations proliferated when SPI was combined with KM and organisational learning initiatives. In Baskerville and Pries-Heje (1998) an attempt to directly incorporate KM concepts into the CMM model is described; these authors suggest an approach to control and measure KM key process areas and, as such, describe assessable measures to an organisation's KM capabilities. Mathiassen and Pourkomeylian (2003) describe how KM and the choice of KM strategy is an important factor when planning SPI initiatives and show how it is beneficial to continuously balance KM strategies with relation to the organisation's maturity and the actual SPI effort. This emphasis on knowledge in SPI underlines that SPI - with the KM efforts conducted - in IT and software development organisations in fact is KM! Most SPI approaches implicitly recognise the processes of establishing organisational learning. As mentioned earlier, the top level of the CMM is labelled 'optimised' and focuses on the continuous task of experience based improvement (Paulk *et al.*, 1995). A level 5 organisation by definition is a learning organisation immersed in double loop learning, and the focus of the SPI initiatives in such an organisation is to facilitate learning activities by establishing measurement and feedback mechanisms, qualitatively and quantitatively. This reinforces that KM is a requirement for SPI.

5 Combining Archetype and Approach – a Balanced Theory of KM in SPI

Organisations which are interested in implementing a KM-based SPI initiative need to consider what kind of approach they want to follow when combining KM with SPI. Based on our conceptual analysis we propose a balanced theory of KM in SPI which might support choosing an appropriate approach. We suggest that that there is a strong correlation between the SPI approach and the knowledge organisation archetype. We have introduced the distinction between exemplary and situational knowledge organisation archetypes and utilise this distinction in our elaboration of the proposed theory of KM in SPI. As a first step, we argue that KM-based SPI activities are balanced along two end points of a continuum: the organisational knowledge type (exemplary vs. situational) and the choice of SPI approach (normative vs. reflective) where knowledge organisation archetypes and SPI approaches closely relate and support each other: theories from the KM literature can be applied as a way to determine which type of SPI approach might fit in a specific organisation — or a specific scenario. Or alternatively—the characteristics of an applied SPI approach can tell which KM approach(es) will constitute a best fit for the SPI paradigm already present. As a consequence, to choose or adjust to a specific SPI approach requires taking the knowledge organisation archetype into consideration. A strict normative SPI approach will be harder to apply in a software organisation which primary business is to deliver new and conceptually different highly customised solutions which result from a situational knowledge organisation. Likewise, a reflective very adaptable SPI approach might be inefficient in an exemplary knowledge organisation since a norm based standard solution might be applicable with only minor adjustments — and most likely also better supporting the business. If the

prevailing knowledge organisation in a company is of the exemplary type, the KM strategy in the organisation is usually anchored in the formal documentation requirements as prescribed by e.g. the CMM. Such an approach relies on a codification strategy to knowledge sharing in which documents act as the prime source of explicated knowledge. According to our theoretical considerations above such a setup would best be supported by a norm based approach to SPI. If the prevailing knowledge organisation is of the situational type a personalisation strategy strongly relying on tacit knowledge is usually implemented. According to our theoretical considerations such a setup would best be supported by a reflective approach to SPI. Figure 3 depicts these basic relations between knowledge organisation archetypes and SPI approaches.

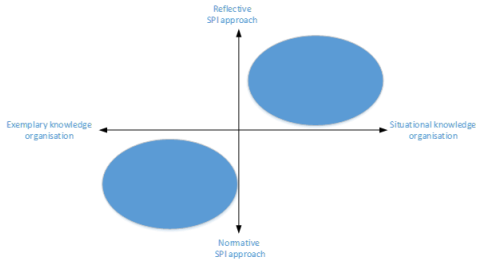

Figure 3: The Relation between Knowledge Organisation Archetypes and SPI Approaches

However, if an organisation practices a norm based approach to SPI, in which the underlying model, e.g. the CMM, prescribes which improvement areas are of relevance to rise in maturity, viz., to comply with higher levels of the CMM, this approach may not fully satisfy the SPI requirements of the organisation. Other important improvements may be relevant which were not prescribed by the norm. In this case the normative approach benefits from being supplemented with an alternative approach. Whereas such an organisational setting might meet the requirements of traceability, documentation, and formalism required by the norm (the CMM model) and might suit management well, a problem arises when the organisation is not able to leverage its knowledge sharing from a project level, which is usually the operational level in software companies, onto an organisational level in an efficient way. To overcome this problem, a solution might be to change the project evaluation process to rely on a personalisation strategy for knowledge sharing. This assures the transfer of the highly complex and contextual information that constitutes the experiences, which were gained in projects, but are useful for other parts of the organisation. Such a solution presents a problem oriented, reflective approach as it originates from perceived problems in the organisation and suggestions from the KM field. It indicates how a KM-based approach to identify SPI improvement areas can be beneficial to the identification and design of new and improved processes. It shows however also that the relationship between knowledge organisation archetypes and SPI approaches is more complex than the conjunctive relation depicted in figure 3. It points to a more nuanced balanced theory of KM in SPI where an exemplary organisation that utilizes a highly normative approach benefits from moving into a more reflective, situational mode, and where a situational organisation that utilizes a highly reflective approach can benefit from moving into a more normative, exemplary mode. This is depicted in Figure 4. This proposed theory offsets the choice of SPI strategy. It balances the adaption of a normative model which prescribes which areas are relevant to improve with a reflective approach which is anchored in actual problems experienced in the organisation and supplemented by suggestions from theory. It also balances the prevailing knowledge organisation type which has an impact on the organisation's ability to collect and share experiences on several organisational levels with its counterpart reflecting the characteristics and advantages of the exemplary and situational knowledge archetypes. The balanced theory of KM in SPI takes into account and reinforces the importance of critically assessing an organisation's current SPI approach when further developing process improvements in an IT and software organisation. The existing paradigm affects the portfolio of possible improvements. A normative approach cannot identify improvement areas 'outside' the chosen model. It can provide a 'complete' and consistent approach to plan a large improvement effort with regard to the model. A reflective approach provides specially designed solutions, often to actual problems. However, it does not combine these into a larger framework which coordinates the SPI effort. The proposed theory acknowledges that a correlation between the normative approach and the exemplary organisation, and similarly between the situational organisation and the reflective approach exists. Most normative approaches are based on an underlying rational model which describes how a software organisation functions. Therefore, within a normative paradigm, an exemplary knowledge organisation, implementing a codification based approach to KM, is preferred. In this setting the underlying model can be used to decode codified information into usable knowledge. On the other hand the situational knowledge organisation supports a reflective approach to SPI. In an organisation

relying on a reflective approach to conduct SPI a situational knowledge organisation archetype provides the details and context to support appropriate reflections to understand of the specific problems of the organisation.

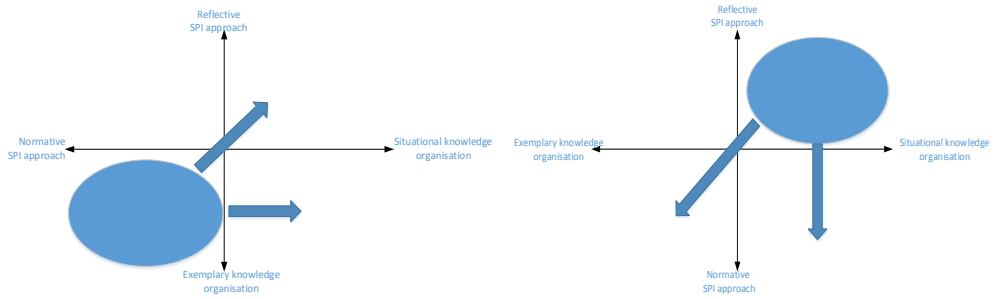

Figure 4: A balanced theory of KM in SPI

Each approach has its particular advantages and disadvantages. A normative SPI approach contains a road-map which makes it easier to apply the approach in an organisation. The norms prescribed by the approach are based on best practices from other organisations. This makes it possible to re-use these practices and in this way achieves faster results. On the other side a normative approach is limited to present only one model under the assumption that this model fits all organisations and improvements, and therefore might only partly fit the organisation in which it is implemented. This might result in the implementation of irrelevant initiatives or in implementing initiatives based on false premises. A reflective approach to SPI is customised to precisely the organisational setting in which it is implemented, and only includes those initiatives which are of relevance in a given situation. It is therefore oriented towards a step-by-step implementation. On the other hand a reflective approach requires an effort to analysis and to customise it to the setting in which it is implemented. The reflective approach does not secure a coherent approach to SPI which might reduce the synergies from several initiatives that are not coordinated. Without a model to fall back on and thus faced with a certain complexity of a unique problem and solution such an approach is harder to control. The exemplary knowledge organisation provides fast and efficient knowledge sharing capabilities, but is not handling sudden and potentially disruptive changes in the organisational environment well. In contrast, the situational knowledge organisation is highly adaptable to changes, but requires more investments to continuously analyse the current settings. The proposed balanced theory embraces the necessity for organisations to reconcile the different requirements and to create a setup in which the prevailing knowledge organisation type is aligned with the primarily applied SPI approach. It also takes into account that an organisation needs to, and can find the combination that balances the advantages and disadvantages of either of the approaches in a way that best fits the organisation's requirements. It allows organisations to, for example, implement a normative approach while introducing specifically tailored processes based on a particular problem situation, and thus shifting the balance of the knowledge organisation archetype from an exemplary towards a situational one.

6 Conclusions

We presented the conceptual basis for investigating how software companies can strengthen their process improvement activities through knowledge management. We answered the research question: *What is the conceptual relationship between KM and SPI?* through a study of the knowledge management and software process improvement where we identified two knowledge organisations archetypes which we labelled exemplary and situational knowledge organisations and two approaches to software process improvement, which we called normative and reflective approaches. We conclude that SPI is KM and propose a balanced theory of knowledge organisational archetypes and process improvement approaches. We argue that such a theory can be used as a framework for KM-based improvement initiatives in software organisations. More work to empirically back our proposal is necessary; we have started to do so and have undertaken a research study in a large software organisation which supports our proposition (Hansen, 2009). Further studies will hopefully confirm the proposed balanced theory beyond organisations in the IT and software sector.

7 References

Aaen, I., Bøttcher, P., and Mathiassen, L. 1998. "The Software Factory: Contributions and Illusions," in Baets, W. R. J. (ed), *Proceedings of the 6th European Conference on Information Systems (ECIS)*, (vol. 2), pp 736-750

Aaen, I., Arent, J., Mathiassen, L., and Ngwenyama, O. 2001. "A Conceptual MAP of Software Process Improvement," *Scandinavian Journal of Information Systems*, (13), pp 81–102.

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*.

Argyris, C., and Schön, D. 1974. *Theory in Practice: Increasing Professional Effectiveness*. San Francisco, USA: Jossey-Bass.

Bach, J. 1994. "The Immaturity of the CMM," *American Programmer*, (7:9), pp 96–98.

Basili, V., Caldiera, G., and Rombach, H. D. 1994. "The Experience Factory," in Marciniak, J. J. (ed), *Encyclopedia of Software Engineering*. West Sussex, UK: John Wiley & Sons, Inc.

Basili, V., and Caldiera, G. 1995. "Improve Software Quality by Reusing Knowledge and Experience," *Sloan Management Review*, (37:1), pp 55–64.

Baskerville, R., and Pries-Heje, J. 1998. "Managing Knowledge Capability and Maturity," in Larsen, T. J., Levine, L., and DeGross, J. D. (eds), *Proceedings of the IFIP WG 8.2 and 8.6 Joint Working Conference on Information Systems: Current Issues and Future Changes*, pp 175-196.

Bjerknes, G., L. Mathiassen, L. 2000. "Improving the Customer-Supplier Relation in IT Development," in *Proceedings of the 33th* Hawaiian International Conference on Systems Science.

Bilotta, J. G., and McGrew, J. F. 1998. "A Guttman Scaling of CMM Level 2 Practices: Investigating the Implementation Sequences Underlying Software Engineering Maturity,"*Empirical Software Engineering*, (3:2), pp 159– 177.

Börjesson, A., and Mathiassen, L. 2003. "Making SPI Happen: the IDEAL Distribution of Effort," in *Proceedings of the 36th* Hawaiian International Conference on Systems Science.

Charitou, C. D., and Markides, C. C. 2003. "Responses to Disruptive Strategic Innovation," *MIT Sloan Management Review*, 44(2), pp 55–63.

Christensen, C. M., Johnson, M. W., and Rigby, D. K. 2002. "Foundations For Growth," *MIT Sloan Management Review*, (43:3), pp 22–31.

Davenport, T. H., and Prusak, L. 1998. *Working Knowledge - How Organizations Manage What They Know*. Boston, USA: Harvard Business School Press.

Deming, W. E. 1982. *Out of the Crisis*. Cambridge, Mass.: Productivity Press.

Diaz, M., and Sligo, J. 1997. "How Software Process Improvement Helped Motorola," *IEEE Software*, (14:5), pp 75–81.

Dingsøyr, T. 2001. *Knowledge Management in Medium-Sized Software Consulting Companies*. PhD Thesis. Trondheim: Unipub Publisher.

El-Emam, K., Goldenson, D., McCurley, J., and Herbsleb, J. 2001. "Modelling the Likelihood of Software Process Improvement: An Exploratory Study," *Empirical Software Engineering*, (6:3), pp 207–229.

Hansen, M. T., Nohria, N., and Tierny, T. 1999. "What's Your Strategy for Managing Knowledge?," *Harvard Business Review*, (77:2), pp 106–116.

Hansen, B., Rose, J., and Gitte Tjørnehøj, G. 2004. "Prescription, description, reflection: the shape of the software process improvement field," *International Journal of Information Management*, (24:6), pp 457-472.

Hansen, B. 2009. *Improving Software Process Improvement with Knowledge Management*. *PhD Thesis*. Department of Informatics, Copenhagen Business School, Denmark.

Hedberg, B. 1981. "How Organizations Learn and Unlearn," in Nyström, P. C., and Starbuck, W. H. (eds), *Handbook of Organizational Design,* (vol.1), pp 3-37. Oxford: Oxford University Press.

Humphrey, W. S. 1989. *Managing the Software Process*. Reading, MA, USA: Addison Wesley.

Humphrey, W. S., Snyder, T. R., and Willis, R. R. 1991. "Software Process Improvement at Hughes Aircraft," *IEEE Software*, (8:4), pp 11–23.

ISO/IEC 15504-1. 2004. *Information technology - Process assessment - Part 1: Concepts and vocabulary. Geneva, Switzerland:* ISO Central Secretariat.

Johnson, D. L., and Brodman, J. G. 1996. "Realities and Rewards of Software Process Improvement," *IEEE Software*, (13:6), pp 99–101.

Kautz, K., and Thaysen, K. 2001. "Knowledge, Learning and IT Support in a Small Software Company," *Journal of Knowledge Management*, (5:4), pp 349– 357.

Kautz, K., Thaysen, K., and Vendelø, M. T. 2002. "Knowledge Creation and IT Systems in a Small Software Firm," *OR Insight*, (15:2), pp 11–17.

Kautz, K., and Nielsen, P. A. 2004. "Understanding the Implementation of Software Process Improvement Innovations in Software Organizations," *Information Systems Journal*, (14:1), pp 3–22.

Kuvaja, P., Similä, J., Krzanik, L., Bicego, A., Saukkonen, S., and Koch ,G. 1994. *Software Process Assessment and Improvement—The Bootstrap Approach*. Blackwell Publisher.

Kuvaja, P., Palo, J., and Bicego, A. 1999. "TAPISTRY—A Software Process Improvement Approach Tailored for Small Enterprises," *Software Quality Journal*, (8:2), pp 149–156.

Levinthal, D., and March, J. G. 1993. "The Myopia of Learning," *Strategic Management Journal*, (14:Winter Special Issue), pp 95–112.

Liebowitz, J., and Beckman, T. 1998. *Knowledge Organization*. Boca Raton, USA: St. Lucie Press.

Lyytinen, K., Mathiassen, L., and Ropponen, J. 1998. "Attention Shaping and Software Risk–A Categorical Analysis of Four Classical Risk Management Approaches," *Information Systems Research*, (9:3), pp 233 - 255.

March, J. G. 1991. "Exploration and Exploitation in Organizational Learning. *Organization Science*," (2:1), pp 71–87.

Mathiassen, L., and Pourkomeylian, P. 2003. "Managing Knowledge in a Software Organization," *Journal of Knowledge Management*," (7:2), pp 63–80.

Mintzberg, H. 1990. "Strategy Formation: Schools of Thought," in Fredrickson, J. W. (ed), *Perspectives on Strategic Management*. New York, USA: Harper Business.

Ngwenyama, O., and Nielsen, P. A. 2003. "Competing Values in Software Process Improvement: An Assumption Analysis of CMM from an Organizational Culture Perspective," *IEEE Transactions on Engineering Management*, (50:1), pp 100–112.

Nielsen, P. A., and Kautz, K. (eds). 2008. *Software Processes and Knowledge*. Aalborg, Denmark: Software Innovation Publisher.

Paulk, M. C., Weber, C., Curtis, B., and Chrissis, M. B. 1995. *The Capability Maturity Model: Guidelines for Improving the Software Process*. Reading, MA, USA: Addison Wesley.

Polanyi, M. 1966. *The Tacit Dimension*. London, UK: Routlege and Kegan.

Quinitas, P., Lefrere, H., and Jones, G. 1997. "Knowledge Management: A Strategic Agenda," *Long Range Planning*, (30:3), pp 385–391.

Quinn, J. B., Anderson, P., and Finkelstein, S. 1996. "Managing Professional Intellect: Making the Most of the Best," *Harvard Business Review*, (74:2), pp 71-80.

Saiedian, H., and Chennupati, K. 1999. "Towards an Evaluative Framework for Software Process Improvement Models," *Journal of Systems and Software* , (47:2-3), pp 139–148.

Shapiro, C., and Varian, H. R. 1999. *Information Rules - A strategic Guide to the Network Economy*. Boston, USA: Harvard Business School Press.

Swan, J., Scarbrough, H., and Preston, J. 1999. "Knowledge Management? The Next Fad to Forget People," in Pries-Heje, J., Ciborra, C., Kautz, K., Valor, J., Christiaanse, E., Avison, D., and Heje, C. (eds), *Proceedings of the 7th European Conference on Information Systems (ECIS-99), Copenhagen, Denmark*, pp 668-678.

Walsh, J. P., and Ungson, G. R. 1991. "Organizational Memory," *The Academy of Management Review*, (16:1), pp 57–91.