12 MANAGING KNOWLEDGE CAPABILITY AND MATURITY

Richard Baskerville
University of Georgia
U.S.A.

Jan Pries-Heje
Copenhagen Business School
Denmark

Abstract

Knowledge management is used as underlying theory to develop a set of key process areas for a supplement to the CMM in small or medium sized enterprises (SME) that develop software. These processes involve a focus on managing knowledge capability rather than traditional project management. A longitudinal case study provides evidence that current practices have already established the feasibility of these key process areas.

Keywords: Information technology innovation, knowledge management, technology transfer.

Introduction

Software development is an increasingly important activity in a rapidly increasing number of companies. Software is produced both as a product in itself and embedded in hardware such as electronic equipment, telecommunications, and consumer electronics. Traditional hardware producers often find that software development takes over more and more of the total development cost—often more that 50%—and that software in many cases accounts for most of the functionality of the products.

However, the discipline of software development is immature. Schedule and budget overruns are typical; low quality and functionality never delivered are other typical signs of immaturity. Best practices have been reported for a number of years and, 10 years ago, the capability maturity model (CMM) was introduced.

The Software Engineering Institute (SEI) in Pittsburgh was asked by the U.S. Air Force in 1986 about a systematic way to evaluate software contractors. A study group was formed at SEI, and together with the MITRE Corporation this study group developed a questionnaire with over 100 questions that could be used to divide successful software development contractors from unsuccessful.

The 100 questions worked well as a tool for evaluating the individual contractor, but not for comparing several bidders for a contract. Therefore the questions were divided into a number of groups (now called key process areas) and the groups of questions were assigned to a level. The levels chosen were inspired from a framework developed by Crosby (1979), and the general improvement idea build into the model came from work by Juran and Gryna (1988) and Deming (1982). The resulting model was called the CMM. The CMM is a framework characterizing a path for software process improvement. The path describes key practices at each of five levels. The description includes a number of goals at each level. An organization has to meet the goals at one level to reach the next. In Figure 1, the five maturity levels are shown together with characteristics for the levels and the challenges (key process areas) one is facing to reach the next level.

Maturity levels are composed of one or more key process areas. If the organization develops these areas, the software process is known to improve. "Each key process area

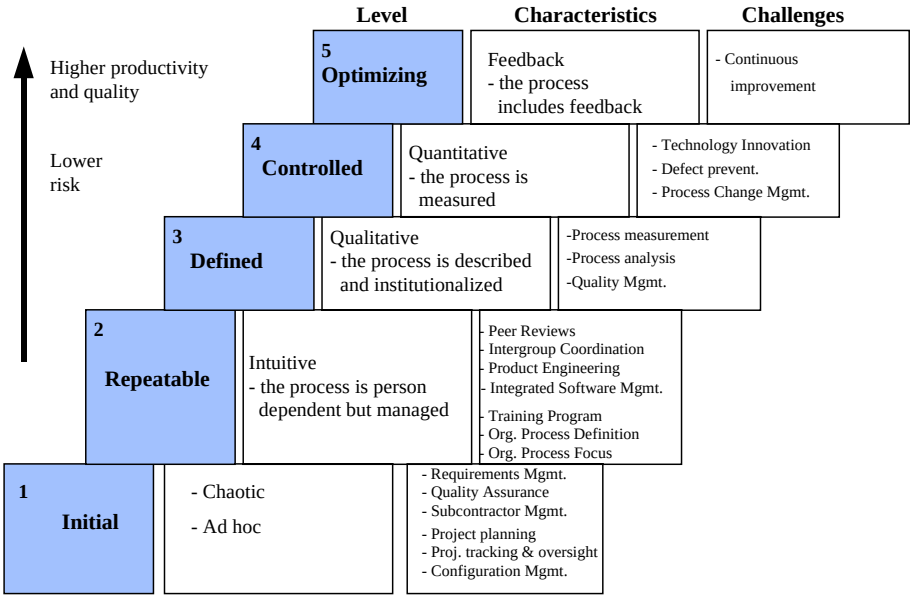

Figure 1. Capability Maturity Model with Five Levels

identifies a cluster of related activities that, when performed collectively, achieve a set of goals considered important for enhancing process capability" (Paulk et al. 1995, p.32). The areas are not considered a complete set, but are known to be key determinants of process capabilities. These areas form not only activities, but also criteria that measure achievement of a maturity level. These criteria are cumulative, meaning that an organization at level 3 will meet the criteria of levels 2 and 3.

The CMM arises because software development is a tumultuous human process. It entails fast-moving computer technology. It entails teams of highly skilled, extremely mobile professional workers. These workers must apply creativity and innovation in their development. We often add to this recipe goals that involve high-stake strategic information systems, and even throw in some rigid deadlines. Without careful management, the process will quickly disintegrate into near-chaos. This chaotic setting prevents software development organizations from delivering their products on time and within budget, if the products are successfully delivered at all.

The CMM ascribes this typical setting as characteristics of an organization at "level 1," or initial maturity, on the five steps of software process management maturity. Industry estimates suggest that three-quarters of all software development organizations exist at this level of maturity (Yourdon 1998). The empirical footing of the CMM is attractive, strongly supporting claims that it reasonably describes actual ways in which software development firms improve (Humphrey 1987). However, with so few real-world examples moving beyond level 2, the empirical data are indicative rather than descriptive.

A number of fundamental reasons underlie our desire to wring order from this chaos. Software development management needs to be able to predict the outcomes in terms of delivery dates and functionality. The organization wants to repeat its successes, and avoid repeating its failures. The organization may also want to improve the quality of its software products and the efficiency of production processes (Paulk et al. 1993).

The fundamental way in which management consistently approaches such problems is by adding structure, even though they may have a flawed understanding of the philosophical assumptions implied by this approach (Truex and Klein 1991). They can improve the structure in the software development process itself, typically by improving or adding methodology, (e.g., Rumbaugh et al. 1991). Alternatively they can improve the structure in the management of the software development process. The CMM focuses on this latter structure, as do alternative software process improvement schemes, such as SPICE, the new ISO standard.[1] To a degree, the CMM is an outcome of the recognition that projects that properly apply the finest software engineering methodology will fail unless the project management succeeds. In a manner similar to total quality management (TQM), software process improvement schemes like SPICE and the CMM bring the benefits of precision engineering into project management.

There are questions about the impact of the CMM on innovation and creativity. The CMM proponents argue that the CMM enables innovation by removing some of the process-based impediments for exploring-designers, giving them time to think, rather

---

[1]See technical report ISO/IEC TR 15504-2: 1998 (E). This report is available from the ISO Central Secretariat, 1 rue de Varambe, CH-1211 Geneve 20, Switzerland, or your national member body of ISO.

than inhibiting them with a chaotic environment that limits their time and opportunities to exchange ideas (Curtis 1994). But critics believe that the CMM structures may encourage a rigid bureaucracy that can stifle creativity and innovation, and demoralize the workforce. Highly competitive and innovative software developers, exemplified by "Borland, Claris, Apple, Symantec, Microsoft and Lotus" did not plunge into the CMM along with the early adopters (Bach 1994b).

> Now the SEI argues that innovation is outside of it scope and that the CMM merely establishes a framework within which innovation may more freely occur...nothing could be further from the truth. Preoccupied with predictability the CMM is profoundly ignorant of the dynamics of innovation.... Because the CMM is distrustful of personal contributions, ignorant of the conditions needed to nurture nonlinear ideas, and content to bury them beneath a constraining superstructure achieving level 2 on the CMM scale may very well stamp out the only flame that lit the company to begin with. [Bach 1994b, pp. 16-17]

There are also questions about the relevance of the CMM for smaller organizations. Small software developers have reported difficulties in applying the CMM for the software process improvement efforts (Johnson and Brodman 1997). In the case of the CMM, its structures have a large-scale, military orientation that may be unsuitable for smaller-scale commercial products developed by the majority of relatively smaller commercial firms (Bach 1994a). This may be because parts of the CMM are irrelevant or problematic for small commercial companies (Herbsleb et al. 1997) while the CMM claims to provide short-term goals; this characteristic horizon reflects the perspective of large organizations. CMM-based improvement projects typically take two years to raise an organization one level in maturity (Herbsleb et al. 1997). For many smaller firms, this is a medium-term horizon, and does not address the immediate needs of the organization. The horizon distance may lead to problems in institutionalizing the measurement philosophy of the CMM and maintaining the momentum required for long-term improvement programs (Hollenbach et al. 1997). As a result, CMM "adaptations" have appeared that are oriented toward workforce issues and smaller organizations, including People-CMM (Hefley et al. 1995) and LOGOS (Brodman and Johnson 1997). LOGOS developers concluded that the CMM needed revising to more clearly align it to the resources, structures, and practices of small businesses, small organizations, and small projects (Johnson and Brodman 1997).

For small organizations, further unfortunate side effects of increased management structure include the introduction of more overhead cost. Also, the innovation and scale problems may place constraints on the knowledge development of the professional workforce and, potentially, limits on organizational adaptive abilities. For highly competitive and innovative commercial software companies, particularly small ones, following the CMM for its own sake may lead to the collapse of competitive potential (Bach 1994b).

In this paper, we examine the issues of size, innovation, and the potential side effects. We also suggest an alternative mechanism for managing software development in small or medium sized software enterprises (software SMEs). This alternative mechanism is oriented toward managing software development innovation rather than software development processes. The purpose of this paper is to report several potential key process areas (KPAs) that may be more suitable for software SMEs than KPAs

proposed with an orientation toward large organizations. In particular, these KPAs are oriented toward the management of knowledge and the development of knowledge capability.

Knowledge Management

Improvements in knowledge management promote those "factors that lead to superior performance: organizational creativity, operational effectiveness and quality of products and services" (Wiig 1993, p. xv). The term has evolved in the IS literature through links to at least two important IS concepts previously housed within the boundaries of specialized fields. The first concept regards knowledge-base management within the field of expert systems (e.g., Zeleny 1987). The other concept regards the management of knowledge as an organizational resource, this usage appearing as early as 1989 in the management literature (Adler 1989). A working definition of this broader view of organizational knowledge is "information embedded in routines and processes which enable action." Knowledge is an innately human quality, residing in the living mind because a person must "identify, interpret and internalize knowledge" (Myers 1996, p. 2).

At least five theoretical concepts underpin knowledge management. Two of these evolve from specialized work in information economics: intellectual capital theory, important for valuing "soft" organizational assets in accounting and business law, and knowledge economy theory from specialized research into the economic role of consulting. Three others evolve from organizational strategy research: core competence management, dumbsizing, and knowledge alliances. Strategic IS theory regards knowledge as a fundamental resource that enables organizations to compete more effectively in their markets (Earl 1997).

Competence-based competition sees organizational competencies as a key resource under established resource-based theories of the firm. These theories of modern competition emphasize the importance of organizational "core competencies" (Prahalad and Hamel 1990). Knowledge management is important for practice in this new competitive environment because it is necessary for developing and maintaining organizational core competence (Sanchez and Heene 1997a, 1997b). Dumbsizing refers to rapid, nonlinear change, sometimes disregarding knowledge, which can undermine sustainable profitability. Important knowledge management factors that are sometimes damaged include reduced research and development (R&D), deteriorated teamwork, crippled professional support, and decreased creativity (Eisenberg 1997). Knowledge alliances involve strategic alliances with other firms in order to balance knowledge deficiencies, obtain necessary competencies, or create new knowledge.

Small or Medium Sized Software Enterprises

Small and medium sized enterprises (SMEs) are defined in varying ways (Lauder, Boocock, and Presley 1994). Criteria used in Europe typically include independence, employee population, and annual turnover or capitalization and independence. Independence means that SMEs must have less than 25% of their control in the hands

of a large enterprise or jointly held by several large enterprises. SMEs have 10 to 250 employees, an annual turnover of less than 40 million ECU, or an annual balance sheet total of less than 5 million ECU. Estimates of the percentage of European firms that fall into this category vary from 6% to 9% (cf. de Koning and Snijders 1992; Mulhern 1995). In contrast, larger firms make up 0.1%, and the remaining firms are very small (micro) enterprises.

Limits on capitalization and lack of liquidity are often considered to be one of the major threats to survival of SMEs (de Koning and Snijders 1992; Rovere 1996; Theng and Boon 1996). The layering of expensive, formal organizational structures such as those involved in large-scale software CMM is not an attractive strategy in such circumstances, particularly if those structures might be associated authoritarian supervision or alienation (Phifer 1978). There are some indications that SMEs avoid rigid organizational structures. For example, SMEs have not adopted TQM to the same extent as larger organizations (Ghobadian and Gallear 1996). Also, SMEs excel in unstable environments and seem better able to flexibly adapt to fast-paced changes in technology and competition than their large-firm competitors. Not surprisingly, SMEs are a wellspring of innovation, both in terms of patent applications (Corsten and Lang 1988) and number of innovations per employee (Rovere 1996).

Studies of the relationship between organizational size and employee turnover have led to mixed or indifferent results (Benson, Dickenson, and Neidt 1987; Tyler 1986). SMEs in general face the same problems as larger and smaller organizations. For software developers, there is a severe problem with employee turnover and mobility, with some estimates as high as 40% turnover (Hein 1998). Like their larger competitors, SME software developers leave for more salary, more experience, more interesting and challenging work, *etc.* Interestingly, developers in companies with fewer than 50 employees are prone to leave when the company begins to grow too large (Garden 1992).

Software SMEs are at a particular disadvantage with regard to developer turnover because they have reduced access to technological information, which places limits on developer innovation (Rovere 1996). Professional employees thrive on the latest information and knowledge, which they know to be essential for their sustained success. Meyer (1997) called this the "FMS ('fraid of missing something) syndrome." There is an essential career need to stay on the cutting edge of technology, and workers verify this achievement by regularly interacting with a network of peers. When this syndrome is correlated with the software SME characteristic of reduced access to information, there are indications of two problems for software SMEs: rapid depreciation of its most valuable commodity (workforce knowledge) and developer turnover that both results from, and exacerbates, the depreciation.

Knowledge is not only a key factor in developer turnover, it is an essential element underlying innovation (Burgelman and Sayles 1986). Professional developers, with their essential knowledge relationships, and knowledge alliances create "fertile fields" for the innovation that software SMEs must have to compete (Kanter 1988). Essentially, unstable software SME knowledge can destabilize innovation, which in turn can drive out developers, further deplete organizational knowledge and bring innovation to a screeching halt. Without innovation, software SMEs lose their central competitive edge in the marketplace.

Methodology

The rationale for the research methodology arises from the exploratory nature of this study: The discovery of concepts in an arena that is not well studied. Originally we were concerned about accommodating highly multivariate, contextual data. However, the rationale shifted toward the need to study the emergence of organizations and the durability of management techniques. These requirements were satisfied by a research design that involved a longitudinal, embedded, single case setting (Yin 1989). The setting is a single case, centered on innovative software development within one small information technology company. Data was collected using focused interviews involving a semistructured case study protocol over a three year period. The interviews were conducted on-site, enabling additional data collection through field-study style direct observation. The interviews and site observations were audio or video taped. The collected data was then transcribed into field notes.

The field notes were then coded interpretively to determine if the organization exhibited characteristics of the software SME problems discussed above. As the case developed over time, the researchers were increasingly drawn to focus on organizational activities that implemented typical knowledge management processes described in the literature, perhaps because the organization was working to maintain their market position by honing their knowledge capability rather than their project management capability. This is consistent with a software SME that recognizes its competitive advantage. The following narrative of our findings describes the early case, and then discusses the activities relevant to their use of knowledge management techniques to improve their innovation. It is important to acknowledge that the subjects' behavior in this case is not intensely driven by theory. The interpretive description that follows superimposes the theoretical importance of their actions. The subjects were concerned about managing their knowledge capability, but they intuitively focused their attention on managing knowledge and invented their strategies perhaps without realizing the degree to which their action adhered to recommended knowledge management practices and established knowledge management theory.

Since our interpretation is framing the reported activities in certain theories, we will open windows into the data to help reveal the degree and depth of interpretation in determining the findings,. The following section provides this interpretation. Following these findings we summarize the general learning in terms from the knowledge management literature and relate this to alternative key process areas for managing knowledge capability.

Case Findings

The software SME in this case is a small Danish company, Proventum. Proventum designs, develops, and operates Internet web sites to support the electronic commerce initiatives of their client companies. The software development grew from an original mission in web advertising and is now the main thrust of the company. Proventum's first customer was a major wholesale supplier of data processing equipment. Proventum created one of the first business-to-business web sites for this client, which catered to about 2,000 equipment retailers and resellers who needed access to pricing and

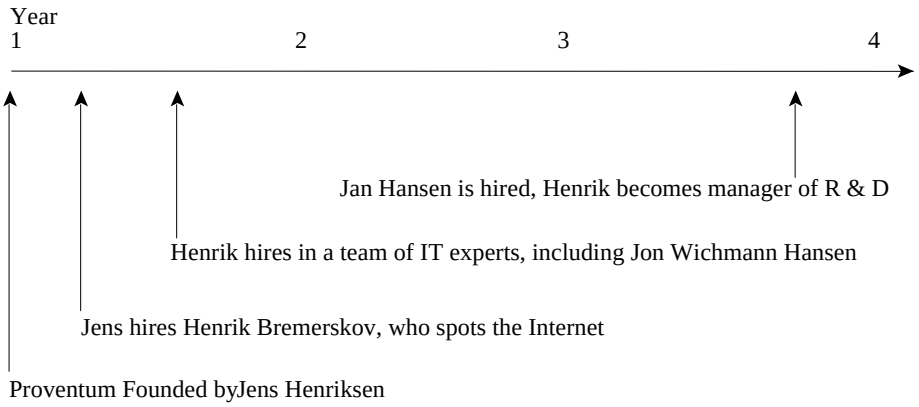

Figure 2. Proventum Subject Time Line

availability information about the many product lines. The site provided dynamic item lists that included the correct price schedules for each retailer and reseller based on their individual volume and contracted discount profiles. Proventum's successful work led to a series of word-of-mouth referrals that brought one new customer after another. (See Figure 2 for a subject time line of Proventum.)

Proventum rapidly grew to 20 employees during its first year, including project managers, designers, programmers, and database specialists. The staff was youthful and voluntarily worked incredible hours of intense and thrilling systems development. They occupied most of one floor in a small office building, and this floor was alive with on-going work during holidays and weekends. They were fiercely proud that they had never missed a deadline, and that once the staff worked round-the-clock for four days in order to preserve this record.

Henrik was one of the major drivers that sent Proventum into the electronic commerce marketplace. He spotted the commercial potential of the Internet craze in its infancy. He characterizes Proventum as a "modern solution provider using the web as its platform for solutions." Proventum helps its customers focus on how their business needs can be satisfied through the new Internet media. Proventum then helps define the requirements, develop the site, and usually continues to operate and maintain the site after its development. Sites are usually developed "on-line" on the Internet so that its customers can watch, evaluate, and redirect the construction of the site.

Proventum sites typically engage cutting-edge web technology to support advanced "business-to-business" applications, meaning that the pages were designed for use by the business partners of the site-owners. The sites were not intended to be an "institutional" public-relations presence, a simple information resource, or a retail catalog. Instead Proventum sites were a versatile and convenient communications utility for the use of customers.

Proventum drew on IT for software programming, database technology, Internet technology, and web technology. Several early Proventum solutions involved making organizational data, e.g., lists of product characteristics, available to customers via the Internet. This process was innovative and complex, since these were the days before the

database middleware simplified such applications. Proventum has historically pushed the technical envelope of the Internet for its clients.

During the first three years, the sources of the IT knowledge within Proventum were informal. The database technology seems to have sprung from the university-level education of one or two key employees. Knowledge about the business needs of the customers came from the experience that Henrik and Jens (the original Proventum owner) gained from previous careers in consumer-oriented advertising and market analysis. Most of the Internet and web knowledge was either brought in by new employees or developed informally by the individual workers after joining Proventum. In the former case, the new employees previously had gained knowledge of the Internet and web technology through personal interests, reading about, experimenting with, or exploring the Internet out of curiosity. In the latter case, this curiosity had driven the individuals to experiment with new Internet technology in a problem-solving mode while looking for solutions to customer requirements. Often, the individuals discovered solutions by regularly "surfing" the web and visiting others' web sites that frequently triggered new ideas for Proventum customers.

During the first two years, Proventum's body of knowledge could be divided into shared knowledge or compartmented knowledge. Shared knowledge included Internet programming languages like HTML and Perl. Compartmented knowledge was vested in individual experts. For example, one was a specialist in cgi programs, another was the network expert, and a third was the authority on programming. Proventum seemed too small for any cross-training or formal classes. For example, if a solution required a cgi script for a web client, then the task was given to the cgi guru. They didn't bother about documentation, and programming intricacies were not discussed. Proventum was not concerned about disseminating or withholding particular knowledge within the organization, since it seemed everything they knew was available through the Internet, and more-or-less in the public domain. Only in their third year did they begin to recognize the importance of their own innovations in the use of this technology:

> *We survive by being technologically ahead of our competitors, and by being able to visualize more far-reaching and wide-ranging solutions for our customers than our competitors are able to. (Henrik)*

However, during their third year, the typical problems of software development organizations began to threaten the firm's survival. Amid the chaotic development milieu, disputes arose over how to run projects and which technologies were important, and programmers began to quit. The firm dwindled to 12 employees. Proventum seemed unable to repeat or build on its successes:

> *I realized that all the documents needed to support this, namely customer contract, project presentation, budget, and requirements specification, were nowhere, and there were a thousand different meanings within Proventum about how they should look....Today we have as many different contracts as we have employees, because we don't have a template to work from. (Jan)*
>
> *We need to be better at exploiting the knowledge from previous projects, much better, so we don't make the same things again and again and again. (Henrik)*

Proventum considered the CMM, and Jon in particular examined its potential closely. His first reaction was:

> *A characteristic of the first maturity level is the anarchy that you find among the developers. I can nod in recognition to that. (Jon)*

But the six KPAs at level 2 in the CMM model just didn't seem "right" for Proventum. The CMM is credited, however, as part of the inspiration for the following actions that Proventum tried to implement:

1. Systematic estimation of application size
2. Making a requirements specification
3. Writing and following a test strategy
4. Project management using Gantt and PERT diagrams
5. Introduction of a strategy to decide which projects to accept and which ones to reject

Essentially, Proventum lacked the two structural keys to software development: a development methodology and a defined project management process. They clearly recognized structure as the solution to their problems:

> *Nine months ago I felt a need for getting more structure in here.... [The customers] require project management, they require that we use methods, they require structure and so on. (Henrik)*
>
> *It became clear that what they needed was first an overview over activities and the relations between the different elements of the organization, and second trying to define a form of structure... structure was not well defined. It was just: "We need structure," without discussing what structure was. (Jan)*

Proventum also understood how they were suffering from the high turnover among IT employees. Proventum recognized that they were losing important organizational knowledge. That their "largest bottleneck" was their inability to attract and retain "enough qualified employees" (Henrik). Proventum clearly realized that they were in a critical transition stage between a small, unstructured, turbulent start-up and a more stable medium-sized enterprise:

> *There has been an incredible personnel turnover in this company. Jon and Henrik have been here for the longest time, but then there is a 1.5 year jump. The third person hasn't been here longer than a year....I know we need a place where we collect knowledge. Right now we have knowledge everywhere. We have public folders, we have documents, we have everything. I just want to say that what we need is for knowledge to get collected. (Jan)*
>
> *We are in midstream between being a little newly started company, still totally chaotic, and being an established organization with everything within narrow limits. (Henrik)*

Proventum took steps to prevail in this transition. These steps included establishing an organizational strategy that directly addressed these problems, adopting a systems development methodology, and improving their ability to manage their knowledge.

They addressed the need for a methodology by inventing and implementing their own standard development approach. Perhaps it is not surprising that an innovative software SME would choose its own distinctive approach, and were particularly pleased with their use of state-machine diagrams suggested by a newly hired university graduate. Table 1 is an overview of their approach. From a CMM perspective, establishing a software engineering methodology is an essential foundation before beginning work on improving the development management processes.

Table 1. Proventum Software Development Methodology Outline

Phase	**Documentation**
1. Prepare Project Presentation. The Consultants–with technical leads from the technical people and the designers	Project Presentation Requirements Specification and Contract
2. Start Up meeting	Structure Diagrams
3. Prepare plan for Navigation within site (cooperatively with customer)	State-Machine Diagrams
4a. Program Technical Analysis and Programming and setting up backbone (carried out by technical people in parallel with 4b.)	
4b. Design-idea, front page, other pages. Approval by customer. Further work on design of site approval by Customer	
5. Script programming. Integration of backbone and design (+ eventual HTML work)	
6. Approval by Customer	
7. System test	(Approval documents)
8. Maintenance (and User) Manual	Test documentation
9. Hand over to Customer	Documentation

Proventum also explicitly developed a strategy that was linked to their knowledge problems. All managers participated in a weekend retreat to discuss strategy and to determine their primary product focus:

> *A considerable focusing of the organization has happened....We do not get involved in so many new things as we did earlier. (Henrik)*

To a certain degree, Proventum then reorganized introducing new management structures that focused on improving their knowledge management. They strengthened their middle management by hiring Jan,[2] an individual "who has a strong theoretical education and also some organizational understanding." Jan began his work in classic middle management style, looking both at the grand plan and the practical realities,

> *using all his time on interviewing people. How do they like to work here? What are their career plans? What are their education plans, and so on and so forth....He is also looking at the organization as a whole: Do we have the right structure, the right people, how should it look in the future so we can get a more stable organization... when do we need to meet, what kind of summaries should there be from the meetings. (Henrik)*

[2]Quotes are not chronological. Jan's critical reflections on the early history of Proventum have appeared earlier in our descriptions. See the subject time line in Figure 2.

Proventum introduced "official" knowledge management structures. Henrik was designated as manager of research and development, a role he characterized as *"researcher, investigator, discoverer of new things, and technologies, and new ways to do things."* The firm introduced a formal scheme that charged every developer with a responsibility for further research and education, and the responsibility to share this knowledge:

> *each person has four hours per week for what I would call further education....I have a list with 40 issues, and each person has a number of them, where he has to perfect his knowledge....The important areas have more than one person assigned, of course.... Having completed your further education, you are obliged to post it on the net...you send a mail saying, I have found out this and that. (Jon)*

This scheme highlights the potentially broad impact of some knowledge management improvements. Not only does this scheme increase knowledge circulating into and around Proventum, it also enables the professional developers to keep abreast, and to verify the value of this knowledge through communication among a network of the peers. These factors were mentioned earlier as potential ways to improve professional retention, further improving organizational knowledge retention. Indeed, Proventum provided each employee with an ISDN link to their homes, a mechanism that also promotes these employee retention factors. Proventum made a number of changes in their treatment of professional developers that were in line with the recommendations discussed earlier. Jan introduced three documents for each worker:

> *There is of course the legal document, the contract. Then there is the job description, and then there is the education plan, so that the individual employee can see his own goal in being here, where will I be in a year, and it can either be professional or organizational. (Jon)*

Also important from the knowledge management viewpoint, the developers were not constrained by job descriptions that compartmented or confined their role functionally, but Jan and Jon insisted that the various developers' expertise was broadly drawn into basic requirements definitions during the early phases of every project. These broad job descriptions are known to promote creativity and innovation through interaction of people with different professional foci. Proventum probably did not grasp the innovative importance of this breadth, but saw the move in more limited terms:

> *It is primarily related to job satisfaction. Instead of being called a "propeller head" or a "coding monkey," or whatever people call it, they are involved in defining their own task. (Jon)*

Innovation and creativity are key elements in the creation of knowledge, and these improvements also advanced their knowledge management. The organization's development methodology is further evidence of an improved environment for creativity and innovation. The first phase explicitly included the collaboration of both the consulting and application development departments, a structural integration that instituted cross-functional interaction from the earliest stages of a project.

The Proventum knowledge-oriented strategy is also evident in their readiness to engage in interorganizational ventures. Proventum is also very proud of their role in introducing SET credit card transactions into the country, an "epoch-making" event for the company. This involved an alliance between Proventum, IBM, and a credit card clearing house. Supported by this alliance, Proventum developed a web site for the automobile association that pioneered the new technology. This alliance with the other vendors was also a critical knowledge alliance that enabled Proventum developers to

learn about the technology as they initiated its first application: *"I don't think you can say they were customers, one can say we cooperated with them." (Jon)*

Outcomes in reporting case studies are always tenuous. However, at the time of this publication, Proventum still exists, and has continued to prosper.

Knowledge Management Key Process Areas

The activities in the Proventum case are comparable with the literature in knowledge management. This comparison indicates that the learning from this case, when linked to the literature, can be used to suggest generalized KPAs for software SME knowledge capability management as a supplement or adjustment to large-scale mechanisms like the CMM. Beyond noting the fact that Proventum considered and rejected the CMM, this paper does not directly test the relevance of the CMM for software SMEs, since this work already appears in the research literature (Johnson and Brodman 1997). The case interpretation that underlies these conclusions are italicized below.

The knowledge management process is necessarily loose and collaborative because knowledge is recognized to be fuzzy and messy (Allee 1997). It is also a difficult process because the human qualities of knowledge, such as experience, intuition, and beliefs, are not only the most valuable, but are also the most difficult to manage and maximize (Davenport and Prusak 1997). The knowledge management process involves the following two key processes: linkage between knowledge management and organizational goals, and an active organizational, behavioral, and technical knowledge infrastructure (Davenport, DeLong, and Beers 1998). Both of these processes are evident in the Proventum case.

KPA #1, knowledge management should be closely linked to organizational strategy and goals (Davenport, DeLong, and Beers 1998). Indeed, the basic idea is that knowledge strategy will guide substantial parts of the strategy and goals (Earl 1997). The purposes of knowledge management have to be expressed in clear language to guide the development and implementation of knowledge management. One important implication of the linkage to organizational goals is the need to measure and account for the outcomes. The quantitative, quality management measures are problematic for this purpose; however, qualitative assessment frameworks have become available, such as the knowledge management assessment framework (Jordan and Jones 1997). This framework helps assess an organization's knowledge acquisition focus and search style, the location, procedures, activities, and scope of its problem-solving approaches, the breadth and processes of knowledge dissemination, the identity and resource type of knowledge ownership, and representation mode of knowledge storage and memory.

> *Considering the dimensions in Table 2, we find that Proventum strategies indicate a generally positive movement on these qualitative scales. For example, their plans to have each worker take responsibility for researching and disseminating knowledge on key topics improved both dimensions of knowledge acquisition, creating deliberate and focused searches of external sources. Their dissemination scheme formalized the distribution on the organizational intranet, leading to a wide distribution. The plan also improved ownership by increasing the redundancy of technical knowledge and improved storage and memory, since the intranet scheme required an articulation of the newly acquired technical knowledge.*

Table 2. The Knowledge Management Assessment Framework (Adapted from Jordan and Jones 1997)

Superordinate Categories	Dimensions	Scale	
Knowledge acquisition	Focus	internal sources	external sources
	Search	opportunistic	deliberate and focused
Problem solving	Primary unit	individual	team
	Procedures	trial and error	heuristics
	Direction of activities	experiential, hands-on	abstract, representational
	Scope	incremental improvements	radical innovation
Dissemination	Knowledge sharing process	informal discussions	formal meetings or databases
	Breadth	narrow, need-to-know	wide publication
Ownership	Locus of emotional identity	personal identification	collective identification
	Resource dispersal	specialist experts	redundant generalists
Storage and memory	Representation	tacit	articulated

Knowledge alliances are a particularly relevant strategy that motivates management to enter into strategic alliances with other firms in order to balance knowledge deficiencies, obtain necessary competencies, or create new knowledge. Knowledge alliance theory, like competence-based competition, evolves from resource-based theory of the firm (Conner and Prahalad 1996). A strategic decision to correct knowledge deficiencies through alliances is a more subtle decision than merely a make-versus-buy choice. Knowledge deficiency includes the lack of organizational diversity that correlates with "dominant logic" in organizational top management, resulting in a routinized or "customary" management logic that inhibits management adaptation and innovation (Bettis and Prahalad 1995; Prahalad and Bettis 1986). Knowledge alliance theory, like the intellectual capital perspective, also entails a measurement problem. It is necessary to identify knowledge deficiencies within the firm, and knowledge strengths of potential partners and competitors. Benchmarking is a notable approach that has been applied for

solving this problem, identifying within other firms industry best practices that have led to superior performance (Drew 1997). Knowledge alliances are also motivated by interorganizational synergy, the ability for organizations to couple their knowledge competencies, offsetting their knowledge deficiencies, thereby enabling new knowledge creation and diffusion processes (Inkpen 1996).

> *The knowledge alliance evident in the Proventum case arose in their involvement with the innovative use of SET technology for one of their clients. Proventum was pioneering and needed the shared knowledge of the credit card clearing house and IBM to make the new system work. Of course, the other firms also needed Proventum's knowledge about industrial-strength transaction web sites.*

The high turnover among software development professionals is related to a number of issues, such as salaries and professional growth. The ability to maintain professional currency in professional knowledge is one factor that can improve retention of key workers and thereby their knowledge. A plan for education, research, and maintenance of a collegial network is one knowledge management technique that is thought to help overcome this turnover (Meyer 1997).

> *While the managers in the Proventum case were very concerned about employee turnover, they saw salaries as a core issue. While this was clearly important, other actions that Proventum management was taking, such as the individual research tasks and the home ISDN connections, were also sound strategies for improving the turnover.*

KPA #2, a "knowledge organization" should provide part of the practical implementation of knowledge management. Although a "chief knowledge officer" (CKO) may not be necessary, successful knowledge management is usually characterized by a designated individual manager in charge of the knowledge management functions (Davenport, DeLong, and Beers 1998). The action in the knowledge management process begins with the formulation and implementation of strategies for construction, embodiment, distribution, and use of organizational knowledge, plus the basic management functions to monitor and measure the knowledge assets and processes (Quintas, Lefrere, and Jones 1997).

> *As a firm in transition from small to medium-size, Proventum was making a major commitment by appointing Henrik as the director of research and development. This major commitment on the part of organizational management elevated Henrik into a role as the de facto CKO.*

In addition, the organizational context (*i.e.,* the entire organization) may need to be revised to enable effective knowledge management. The ideal knowledge organization has been described as "N-form," in contradistinction to the traditional M-form (Hedlund 1994). Primarily, middle management is a critical layer in knowledge organizations for at least two reasons. One reason is because professional knowledge is usually created, transformed, and articulated in the middle management layer. Another reason is that middle management is needed to resolve the contradictions between the grand designs of top management and the limits placed on those designs by the realities of the organization's primary value chain. Middle management is central to the knowledge creation process by rationalizing top management plans and primary value-adding processes (line management) into a progressive unit. The role of middle management

in the knowledge organization is to transform knowledge across organizational levels. The centricity of middle management in knowledge organizations means that these are neither top-down nor bottom-up, but rather are "middle-up-down" organizations (Nonaka and Takeuchi 1995, p. 127).

> *The activities of Jan illustrate how Proventum was moving in this direction. Jan broke away from the previous Proventum notions about top-down team building and began interviewing every developer. By acquiring an understanding of the developers, the people in the primary value chain, Jan began to mediate the grand plan of top management with the realities of the shop floor. Furthermore, Jan was carefully raising the members of the line process into a middle management culture. The programmers became involved in defining the tasks, basically drawing the professional developers into a middle management role rather than a line role.*

Successful knowledge management focuses more in the realm of organizational behavior (Frappaolo 1998). The behavioral infrastructure centers knowledge creation rather than knowledge storage or transfer. This focus is similar to the centrality of innovation in diffusion of innovation. The reason storage and transfer of knowledge are marginalized is because manipulation of knowledge is an essentially human process that cannot be separated from interpretation and reflection. The transfer and use of knowledge implies a certain degree of innate creation. The innovation process is uncertain, fragile, controversial, and political, a set of conflicting, problematic attributes. Invention and creativity usually involves nonconformist thinking, which raises a social struggle, and both the process and the result inevitably reach across clear boundaries, which raises a political struggle. Since innovation is promoted by "cross fertilization" of ideas, structural integration and broad job definitions within organizations can promote innovation by lowering political barriers and exposing contrary premises (Kanter 1988).

> *Within the Proventum case, integration and broadened job horizons arose from the involvement of the developers in formulating presentations and writing specifications. The developers and the consultants interacted regularly with each other, knowledge allies, and the customers. In addition, the research role raised the stature of each developer from that of a knowledge "sink" to that of an official knowledge "source." Each person was responsible for creating knowledge and sharing their ideas, making idea articulation a routine form of behavior among Proventum workers. This plan also broke down barriers and explicated contrary premises thereby further promoting innovation.*

The framework for these software SME knowledge capability KPAs is shown in Table 3. This framework represents the intersection between knowledge management theory and practice, and SME theory and practice, innovation theory, and is illustrated by the longitudinal Proventum case study. While these KPAs are fairly alien to the current version of the CMM, other software process improvement schemes are a little more relevant. SPICE, for example, has two processes within its ORG group, namely ORG 3, "human resource management process," and ORG 4, "infrastructure process," that are relevant to knowledge capability KPAs. However, the focus on knowledge management and knowledge capability are unique to the approach described in this paper.

Table 3. The Knowledge Capability Key Process Area (KPA) Framework

KPA	Process	Goals
1	Knowledge strategy	Goal 1. Knowledge management and organizational strategy are explicitly linked
		Goal 2. Knowledge alliances are planned
		Goal 3. Knowledge development and collegial network for professional workforce are planned
2	Knowledge organization	Goal 1. Explicit organizational knowledge management functions are created and operate properly
		Goal 2. Middle management has defined roles in knowledge management
		Goal 3. Developer tasks are structurally integrated and developer jobs are defined as broadly as possible

The case does not strongly support a total rejection of the CMM or other software process improvement techniques. Proventum found "inspiration" in some of the fundamental ideas. For this reason, it appears that managing knowledge capability through knowledge management makes supplemental KPAs for firms with a competitive strategy that depend heavily on innovation. Like other KPAs, these are available individually for firms that "pick and choose" among the KPAs that seem suitable for their culture. Such firms are typically less concerned about meeting CMM level 2 criteria, and more concerned about generally reducing their chaos and surviving in their marketplace.

Implications and Future Research

This study offers qualitative support for alternative process management areas for software SMEs. These alternatives use knowledge management techniques as a form of knowledge capability management. Interestingly, these techniques achieve some of the benefits of larger-scale techniques, like the CMM, without introducing rigid structures and quantitative measures. It is clear that this work suggests that a focus on knowledge capability management could be important for small and medium sized software firms. Typically we find these firms exist in unstable, perhaps rapidly expanding, environments that require rapid adaption and flexibility. Larger organizations have had problems in the past competing with small firms in markets with these characteristics. One implication of this association with instability is that knowledge capability management may be as closely associated with software development in unstable environments as it is with organization size. This association implies that larger organizations, which need to maximize innovation, perhaps because they exist in an unstable environment, might benefit from knowledge capability management as well as

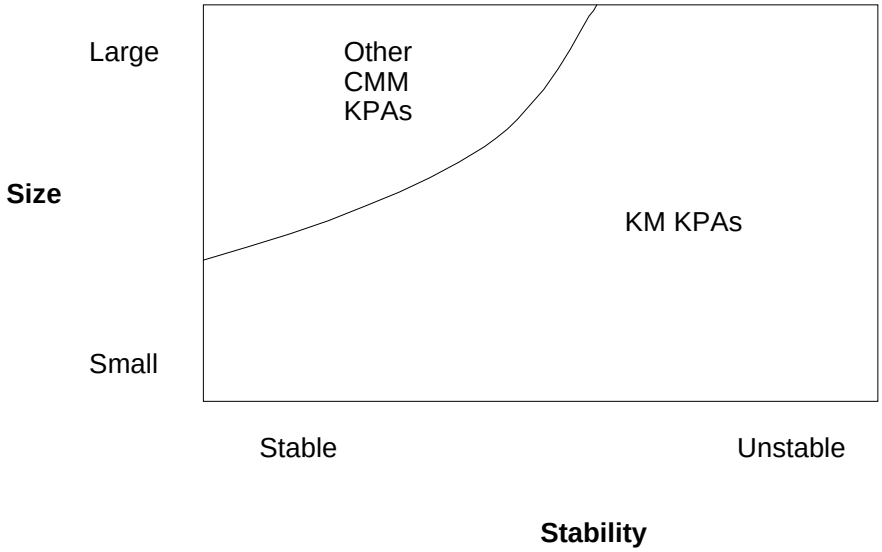

Figure 3. Range of Organizational Settings for Knowledge Capability Management

software process capability management. In such organizations, knowledge management KPAs might profitably supplement or replace other CMM-style KPAs.

When this association is factored together with the tendency for software SMEs to eschew the CMM, we find the implication that innovation management and existing CMM KPAs may be suitable for different organization settings. The existing CMM structure may be most appropriate for large, stable enterprises, while knowledge capability management may be more appropriate for unstable or small organizations. This range of settings is graphically depicted in Figure 3.

Further research is needed to investigate the role of knowledge capability management in large organizations that compete in unstable environments. This additional research might lead to discovery of other KPAs not revealed in the above case. A longitudinal case study approach could be used for such organizations similar to this research. However, it would also be helpful to add evidence from action research in which the KPAs that emerge from this case study are more explicitly applied in a long-term investigation of a software SME or a large enterprise in an unstable environment.

References

Adler, P. S. "When Knowledge Is the Critical Resource, Knowledge Management Is the Critical Task," *IEEE Transactions on Engineering Management* (36:2), 1989, pp. 87-94.

Allee, V. "Twelve Principles of Knowledge Management," *Training & Development* (51:11), 1997, pp. 71-74.

Bach, J. "Enough About Process: What We Need are Heroes," *IEEE Software* (12:2), 1994a, pp. 96-98.

Bach, J. "The Immaturity of the CMM," *American Programmer* (7:9), 1994b, pp. 13-18.

Benson, P. G.; Dickinson, T. L.; and Neidt, C. O. "The Relationship Between Organizational Size and Turnover: A Longitudinal Investigation," *Human Relations* (40:1), 1987, pp. 15-29.

Bettis, R. A., and Prahalad, C. K. "The Dominant Logic: Retrospective and Extension," *Strategic Management Journal* (16:1), 1995, pp. 5-14.

Brodman, J. G., and Johnson, D. L. "Software Process Improvement Approach Tailored for Small Organizations and Small Projects," in *Proceedings of The 1997 IEEE 19th International Conference on Software Engineering*, IEEE, Los Alamitos, CA, 1997, pp. 661-662.

Burgelman, R. A., and Sayles, L. R. *Inside Corporate Innovation: Strategy, Structure, and Managerial Skills*, The Free Press, New York, 1986.

Conner, K. R., and Prahalad, C. K. "A Resource-based Theory of the Firm: Knowledge versus Opportunism," *Organization Science* (7:5), 1996, pp. 477-501.

Corsten, H., and Lang, O. "Innovation Practice in Small and Medium-Sized Enterprises: An Empirical Survey of the Member States of the European Community," *Technovation* (7:2), 1988, pp. 143-154.

Crosby, P. B. *Quality is Free: The Art of Making Quality Certain*, Mentor, New American Library, New York, 1979.

Curtis, B. "A Mature View of the CMM," *American Programmer* (7:9), 1994, pp. 19-28.

Davenport, T. H.; De Long, D. W.; and Beers, M. C. "Successful Knowledge Management Projects," *Sloan Management Review* (39:2), 1998, pp. 43-57.

Davenport, T. H., and Prusak, L. *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, Cambridge, MA, 1997.=

de Koning, A., and Snijders, J. "Policy on Small- and Medium-sized Enterprises in Countries of the European Community," *International Small Business Journal* (10:3), 1992, pp. 25-39.

Deming, W. E. *Out of the Crisis*, MIT Center for Advanced Engineering Study, Cambridge, MA, 1982.

Drew, S. A. W. "From Knowledge to Action: The Impact of Benchmarking on Organizational Performance," *Long Range Planning*, 30 (3), 1997, pp. 427-441.

Earl, M. J. "Knowledge as Strategy: Reflections on Skandia International and Shorko Films," in *Knowledge in Organizations*, L. Prusak (ed.), Butterworth-Heinemann, Boston, 1997, pp. 1-15.

Eisenberg, H. "Reengineering and Dumbsizing: Mismanagement of the Knowledge Resource," *Quality Progress* (30:5), 1997, pp. 57-64.

Frappaolo, C. "Defining Knowledge Management: Four Basic Functions," *Computerworld* (32:8), 1998, p. 80.

Garden, A.-M. "Potential Reasons for Software Employees in Small Companies to Leave Their Present Company," *IEEE Transactions on Engineering Management* (39:3), 1992, pp. 246-253.

Ghobadian, A., and Gallear, D. N. "Total Quality Management in SMEs," *Omega* (24:1), 1996, pp. 83-106.

Hedlund, G. "A Model of Knowledge Management and the N-form Corporation," *Strategic Management Journal* (15/Summer Special Issue), 1994, 73-90.

Hefley, W. E.; Curtis, B.; Miller, S.; and Konrad, M. "People Capability Maturity Model (P-CMM): Incorporating Human Resources into Process Improvement Programs," in *Proceedings of the Annual International Symposium of the National Council on Systems Engineering*, W. W. Schoening and E. E. Barker (eds.), National Council on Systems Engineering, 1995, pp. 559-566.

Hein, K. "Programming Loyalty," *Incentive* (172:2), 1998, pp. 31-34.

Herbsleb, J.; Zubrow, D.; Goldenson, D.; Hayes, W.; and Paulk, M. "Software Quality and the Capability Maturity model," *Communications of the ACM* (40:6), 1997, pp. 30-40.

Hollenbach, C.; Young, R.; Pflugrad, A.; and Smith, D. "Combining Quality and Software Improvement," *Communications of the ACM* (40:6), 1997, pp. 41-45.

Humphrey, W. S. "Characterizing the Software Process: A Maturity Framework," *Technical Report CMU/SEI-87-TR-11*, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 1987.

Inkpen, A. C. "Creating Knowledge Through Collaboration," *California Management Review* (39:1), 1996, pp. 123-140.

Johnson, D. L., and Brodman, J. G. "Tailoring the CMM for Small Businesses, Small Organizations, and Small Projects," *IEEE Software Process Newsletter* (8: Winter), 1997.

Jordan, J., and Jones, P. "Assessing Your Company's Knowledge Management Style," *Long Range Planning* (30:3), 1997, pp. 392-398.

Juran, J. M., and Gryna, F. M. *Juran's Quality Control Handbook*, 4th ed., McGraw-Hill, New York, 1988.

Kanter, R. M. "When a Thousand Flowers Bloom: Structural, Collective, and Social Conditions for Innovation in Organizations," *Research in Organizational Behavior* (10), 1988, pp. 169-211.

Lauder, D.; Boocock, G.; and Presley, J. "The System of Support for SMEs in the UK and Germany," *European Business Review* (94:1), 1994, pp. 9-16.

Meyer, C. "What Makes Workers Tick?" *Inc* (19:18), 1997, pp. 74-83.

Myers, P. S. "Knowledge Management and Organizational Design: An Introduction," in *Knowledge Management and Organizational Design*, P. S. Myers (ed.), Butterworth-Heinemann, Boston, MA, 1996, 1-6.

Mulhern, A. "The SME Sector in Europe: A Broad Perspective," *Journal of Small Business Management* (33:3), 1995, pp. 83-87.

Nonaka, I., and Takeuchi, H. *The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press, New York, 1995.

Paulk, M. C.; Curtis, B.; Chrissis, M. B.; and Weber, C. V. "The Capability Maturity Model for Software Version 1.1," *CMU/SEI-93-TR-24*, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 1993.

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

Phifer, C. B. "How a Small Manufacturing Plant Dealt with a High Employee Turnover Rate," *Industrial Management* (20:5), 1978, pp. 25-28.

Prahalad, C. K., and Bettis, R. A. "The Dominant Logic: A New Linkage Between Diversity and Performance," *Strategic Management Journal* (76:6), 1986, pp. 485-501.

Prahalad, C. K., and Hamel, G. "The Core Competence of the Corporation," *Harvard Business Review* (68:3), 1990, pp. 79-93.

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

Rovere, R. L. L. "IT Diffusion in Small and Medium-sized Enterprises: Elements for Policy Definition," *Information Technology for Development* (7:4), 1996, pp. 169-181.

Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; and Lorensen, W. *Object-Oriented Modeling and Design*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

Sanchez, R., and Heene, A. "A Competence Perspective on Strategic Learning and Knowledge Management," in *Strategic Learning and Knowledge Management*, R. Sanchez and A. Heene (eds.), Wiley, Chichester, UK, 1997a.

Sanchez, R., and Heene, A. "Reinventing Strategic Management: New Theory and Practice for Competence-based Competition," *European Management Journal* (15:3), 1997b, pp. 303-317.

Theng, L. G., and Boon, J. L. W. "An Exploratory Study of Factors Affecting the Failure of Local Small and Medium Enterprises," *Asia Pacific Journal of Management* (13:2), 1996, pp. 47-61.

Truex, D., and Klein, H. "A Rejection of Structure as a Basis for Information Systems Development," in *Collaborative Work, Social Communications and Information Systems*, R. Stamper, R. Lee, P. Kerola, and K. Lyytinen (eds.), North-Holland, Amsterdam, 1991, pp. 213-236.

Tyler, G. W. "Organization Size and Staff Tenure," *Journal of the Operational Research Society* (37:4), 1986, pp. 373-380.

Wiig, K. *Knowledge Management Foundations–Thinking about Thinking–How People and Organizations Create, Represent and Use Knowledge*, Schema Press, Arlington, TX, 1993.

Yin, R. *Study Research: Design and Methods*, Sage, Newbury Park, CA, 1989.

Yourdon, E. "Where's the Basis for Year 2000 Optimism?" *Computerworld* (32:7), 1998, p. 68.

Zeleny, M. "Management Support Systems: Towards Integrated Knowledge Management," *Human Systems Management* (7:1), 1987, pp. 59-70.

About the Authors

Jan Pries-Heje holds an M.Sc. and Ph.D. from Copenhagen Business School, Denmark. He is currently associate professor in Information Systems at Copenhagen Business School. His research interests include information systems development, software engineering, and software process improvement. He is the Danish national representative to IFIP Technical Committee 8 on Information Systems. He was program chair for the first IFIP Working Group 8.6 conference on diffusion and adoption of IT, Oslo, Norway, October 1995. He is conference and organizing chair for the European Conference on Information Systems (ECIS) in Copenhagen, June 1999. E-mail: pries-heje@cbs.dk

Richard Baskerville holds M.Sc. and Ph.D. degrees from the London School of Economics and is associate professor of information systems in the Department of Computer Information Systems of Georgia State University. His research specializes in security of information systems, methods of information systems design and development, and the interaction of information systems and organizations. His interests in methods extends to qualitative research methods. He is an associate editor for *The Information Systems* Journal and *MIS Quarterly.* Baskerville's practical and consulting experience includes advanced information system designs for the U.S. Defense and Energy Departments. He is chair of IFIP Working Group 8.2 and a Chartered Engineer under the British Engineering Council. E-mail: baskerville@acm.org