CRACKING THE CODE: THE DYNAMICS OF PROFESSIONAL KNOWLEDGE

ABSTRACT

Dimensions such as tacit-explicit and personalised-codified have been suggested as vehicles to understand the dynamics of knowledge in the context of professional work. The conventional wisdom is that these underlying perspectives on knowledge are alternatives, inviting either a focus on tacit-explicit or on personalised-codified knowledge. In addition however, the terms 'explicit' and 'codified' tend to be used interchangeably. Here we argue that explicit and codified knowledge focus on distinct and different aspects of professional work and these underlying perspectives should be seen as complementary rather than competing. As a consequence we need to critically rethink the role of codification in knowledge management and on that basis review our understanding of the dynamics of professional knowledge. Drawing upon insights from the systems development profession we propose to distinguish between tacit and explicit knowledge and between informal and codified knowledge as two complementary dimensions of knowledge. This leads to four distinct archetypes of professional knowledge - procedural, community, exemplary, and situated knowledge – and as a consequence to a revised perception of the dynamics involved in knowledge conversion from one type to another. The paper concludes with a critique of the conventional wisdom on knowledge management and with a discussion of the implications of our argument for both research and practice.

Lars Mathiassen
GRA Eminent Scholar and Professor of Digital Commerce
Director, Center for Digital Commerce
eCommerce Institute
J. Mack Robinson College of Business
Georgia State University
PO Box 5029
Atlanta GA 30302-4015
Tel: +1-404-651-0933
Fax: +1-404-463-9292
e-mail: Lars.Mathiassen@eci.gsu.edu

Maxine Robertson & Jacky Swan
IKON Research Centre
Warwick Business School
University of Warwick
Coventry CV4 7AL
United Kingdom
Tel: +44 24 7652 2457
Fax: +44 24 7652 4656
e-mail: **Maxine.Robertson@wbs.ac.uk**
e-mail: **Jacky.Swan@wbs.ac.uk**

INTRODUCTION

The seminal work by Polanyi - *Personal Knowledge* (1962) and *The Tacit Dimension* (1966) has had major impact on both research and practice within knowledge management. It is indeed appealing and straightforward to focus on knowledge that is captured on paper or communicated through some other medium. However in doing so we easily forget other forms of knowledge that are less apparent. Polyani's work is a challenging reminder that knowledge embedded in our practices, largely invisible to us is as, or indeed more, important than knowledge that is readily articulated (Brown & Duguid, 2001). It is therefore well deserved that Polanyi's perspective, focusing as it does on tacit knowledge has had a major influence on recent contributions to Knowledge Management (KM).

The majority of research in the field of KM has been guided by the epistemological distinctions made between tacit and explicit knowledge and the conversion processes between these two types of knowledge. Nonaka and Takeuchi's work is relevant here and is based on experiences from Japanese innovative companies (Nonaka and Takeuchi, 1995). They highlight that tacit knowledge is 'personal', context-specific, and difficult to formalise or communicate. Explicit or 'codified' knowledge, from their perspective, refers to knowledge that is transmittable in formal, systematic language" (p. 59). Based on this understanding they propose four different modes of knowledge conversion: socialisation (from tacit to tacit), externalisation (from tacit to explicit), combination (from explicit to explicit), and internalisation (from explicit to tacit). Another well-cited piece of research in the KM field conducted by Hansen *et al.* (1999) is based on experiences from consulting firms and other professional work contexts and focuses on two distinct KM strategies that firms can choose to adopt. Their research established that in some companies KM strategy centres on the use of IT/IS whereby knowledge is carefully codified and stored in databases to be shared amongst professionals, in others knowledge is shared through networks of interaction. Thus they distinguish between codified and personalised knowledge and suggest "that the choice between codification and personalisation is the central one facing virtually all companies in the area of knowledge management" (p. 107). Knowledge is, in their view, codified to support a people-to-document strategy. That is knowledge is extracted from the person who created it - it is made independent of that person - and subsequently reused for a variety of purposes. In contrast, personalised knowledge, i.e. tacit knowledge - knowledge that has not been codified – is typically shared in brainstorming sessions and one-on-one conversations - a person-to-person strategy.

Both of these approaches to KM are illustrative of the conventional wisdom that the epistemological dimension of knowledge is best understood as a one-dimensional phenomenon. Nonaka and Takeuchi for example, make no distinction between explicit and codified knowledge and Hansen *et al.* use the terms personalised and tacit knowledge interchangeably. This raises important questions in relation to understanding professional knowledge: Can personalised knowledge be explicit? Can tacit knowledge be codified? The conventional frameworks for KM imply a negative answer to both of these questions. In this paper, we focus on the relationships and differences between codified and explicit knowledge in the context of professional work and this leads us to insights that contradict the prevailing dominant perspective. We argue that codification of knowledge plays a crucial role both in relation to maturing professional practices and in relation to the use of information technology to support KM. It is therefore important that people engaged in KM activities appreciate the problems and opportunities related to the codification of knowledge.

In our search for a more robust understanding of professional knowledge we draw upon experiences from the systems development profession. The key challenge in systems development is to change organisations through implementation of modified or newly developed computerised systems (Lyytinen 1987; Mathiassen 1998). Systems development is typically carried out in projects across organisational boundaries. It is a non-routine, knowledge intensive type of activity that involves innovation and collaboration as key elements. Our argument is, in this way, based on a particular type of professional work that has been studied intensively as part of the information systems and software engineering disciplines. Drawing upon insights from this particular professional practice makes it possible for us to create coherence in the argument. We suggest, however, that the findings may have a bearing on professional knowledge more generally.

Our argument is developed as follows. The next section argues why it is important to understand the process of codification in depth as an important aspect of KM practice and the concepts of explicit and codified knowledge are more fully explored. We then propose and describe a two-dimensional understanding of the epistemology of professional knowledge distinguishing between the tacit-explicit dimension and the informal-codified dimension. The next section focuses on the dynamics involved in converting knowledge from one type to the other and we include at this point considerations of the ontological dimension of professional knowledge focusing on individuals, groups, and organisational networks at large. Finally we summarise our critique of the one-dimensional epistemologies (Nonaka 1994; Nonaka & Takeuchi 1995; Hansen *et al.*1999) and discuss the implications of our argument for KM research and practice.

CODIFIED AND EXPLICIT KNOWLEDGE

Polanyi (1962, 1966) has convincingly argued that it is important to understand the tacit and personal dimensions of knowledge. There are, however, a number of reasons why it is equally important to understand codified knowledge in the context of professional work. First, to improve the productivity, operation or quality of outcomes, organisations develop patterns of behaviour that can be reused, to ensure certain work practice disciplines and facilitate socialisation or training of new members. Such patterns can either be embedded in the social practices within the organisation (Schein 1985) or they can be expressed as part of the organisation's procedures and structure (Mintzberg 1983). Organisations tend to develop more patterns as they grow in size and mature, these patterns can increase organisational performance but they can also turn into obstacles for innovation and change (Starbuck, 1992). Second, information technology has become a strategic resource for most organisations (Applegate *et al.* 1999). The computer operates on data, i.e. explicit knowledge or information that is brought into a form that can be read, stored, and processed through the execution of computer programs. To take advantage of the computer's automatic processing capacity, organisations therefore need to explicate knowledge as information and turn information into data (Dahlbom & Mathiassen 1993). They need to explicate and codify knowledge. Third, there are close and intrinsic relations between KM and the use of information technology. The use of information technology supports KM (Alavi & Leidner 2001), but insights from KM can also support and improve professional work related to information technology (Mathiassen *et al.* 2001).

Explicit knowledge stands in contrast to tacit knowledge. It is readily articulated and it is made visible through the use of systems of signs and various media - books, documents, e-mails, and tape recordings all contain explicit knowledge. We also express explicit knowledge

when we speak or engage in other forms of communication with others. Tacit knowledge is in contrast intrinsically related to our behaviour, it is expressed through our actions and it is not readily articulated.

Codified knowledge stands in contrast to informal or unique knowledge. It is shaped according to a code, a convention, or a set of rules. It is generic in that it has to be brought into action each time it is reused, either through the interpretation of codified expressions or through re-enacting generic behavioural patterns. Informal knowledge is in contrast related to unique situations and expressed in a spontaneous manner, manifest in language or practice, independent of specific codes or conventions. Informal knowledge is created in response to unique events whereas codified knowledge is created in response to generic activities.

It is obvious that explicit knowledge always has to be codified to some extent. We cannot explicate knowledge without relying on systems of signs. So explicit knowledge will always be codified on the syntactic level. However, this is not necessarily the case when it comes to contents and the semantic level. For example, an electronic text follows certain syntactic rules, but you can freely express your thoughts as you do in informal, natural language. An electronic text is therefore *explicit but not necessarily codified* beyond the syntactic level. A database, in contrast, contains information that is codified according to the programmer's definition of data items and relations. 'Name' can be codified as 'First name' followed by 'Surname' and both of these can be codified as strings of one or more letters from the alphabet. In this case 'Don Smith' or 'Diana Q' can be stored in the database, but 'D. Smith' cannot be stored as '.' will not be accepted as a letter. A database is in this way *explicit and codified*, thereby supporting certain expressions while rejecting others. The key point here is that beyond the syntactic level, there are important forms of explicit knowledge that are *not* codified. They are unique expressions in some form of natural language. It is also obvious that codified knowledge stored in a computer is explicit. Simply because the very act of programming computers requires explication of both the data to be computed (the database) and of the process to be followed by the computer (the program).

There are, however, other important forms of codified knowledge that are not explicit. Such codes are embedded in social settings and they regulate our behaviour to the extent that we are members of that particular community. These codes are conventions or routines for communication, social interaction, and professional conduct. They are expressed through the ways in which we act and speak and we are seldom conscious of them. However, breakdowns can occur when new members enter the community or when unknown territory has to be explored. The key point is here that codified knowledge is often made explicit and has to be explicated as part of computerised systems. However, there are other important forms of codified knowledge that remain largely tacit and play crucial roles in shaping professional work.

PROFESSIONAL KNOWLEDGE

Let us reconsider the epistemology of knowledge in this light. Rather than seeing explicit knowledge and codified knowledge as synonyms we should see them as related to two different dimensions of knowledge as expressed in Figure 1. The tacit-explicit dimension relates to whether knowledge is visible or readily articulated. The codified-informal dimension relates to whether knowledge is an expression of a code or convention. That leads to identification of four distinct types of professional knowledge that we denote procedural, community, exemplary, and situated. In the following we will examine each of these types in turn through examples from the systems development profession.

Procedural knowledge is codified and explicit and it plays a key role within the systems development profession. We can identify a great variety of forms in which such knowledge appears. Methods are collections of concepts, notations, techniques and tools that prescribe or support certain broad areas of practice like project management or systems analysis and design; a contemporary example of a method is Rational Unified Process (Jackobsen *et al.* 1999). Processes are generic prescriptions of highly focused and specialised activities like configuration management or sub-contract management (Humphrey 1989). Patterns are generalised descriptions of a problem and a related solution, e.g. how to conceptually model customer-client relationships (Coad 1995). In addition there are principles, notations, techniques, templates, standards, and tools. It is indisputable that we find an abundance of procedural knowledge within the systems development profession. This knowledge does play a major role in shaping professional education and training and many organisations invest heavily in the latest or most fashionable procedural knowledge in their attempts to improve performance or gain competitive advantage. There is, however, much debate on the impact that procedural knowledge can have and has on the actual practices of systems development professionals (see for example Mathiassen 1998 for a discussion of that issue).

	Informal	**Codified**
Tacit	Situated knowledge	Community knowledge
Explicit	Exemplary knowledge	Procedural knowledge

Figure 1. Four different types of professional knowledge.

Community knowledge is codified and tacit. Such knowledge is embedded into systems development practices and it can again take various forms. Communities-of-practice are groups of systems developers sharing certain practices that have been developed as a result of joint problem solving activities (Brown & Duguid 1991, 2001; Mathiassen 1998). Such communities-of-practice emerge as new types of applications have to be developed, as new technologies are introduced, or when the contractual conditions for systems development change. They can be small or encompass entire departments. Formative contexts are similar to communities-of-practice in that they support joint problem solving. A formative context is a set of institutional arrangements and cognitive imageries that inform the actor's practices and reflections (Ciborra & Lanzara 1994; Mathiassen 1998). They also shape the way in which the actors think about practices and they can serve as greenhouses in which experiments for organisational change take place and new practices emerge. Tensions therefore easily occur between existing formative contexts (based on established work routines) and emerging formative contexts (based on new approaches or technologies). Community knowledge corresponds in general to what Schein (1985) denotes as the basic assumptions of an organisational culture or to what Argyris and Schon characterise as theories-in-use as opposed to espoused theories. It is taken for granted, seldom expressed, but it still dominates the daily practices within a professional culture.

Exemplary knowledge is informal and explicit. Such knowledge is continuously expressed during project working as professionals participate in episodes, experiments, or produce results. It will be shared with others if it is conceived to be unique or interesting. Exemplary knowledge is thus expressed in project documentation, project group meetings, in department seminars, in memo's and reports, or simply as part of the ongoing story telling between professionals. Exemplary knowledge reflects something new or different or it may be a shared frame of reference that communicates important aspects of a group's history. The experience factory (Basili 1989) is an ambitious attempt to organise and manage useful experiences – exemplary knowledge - so others can reuse. Examples of elements that could be shared are project plans, software specifications, models of human-computer interfaces, and particular software code. On a less ambitious level, systems developers often cut-and-paste to reuse parts of previous documents, they reuse parts of programs, and they access program libraries to take advantage of available resources.

Situated knowledge is informal and tacit. It is intrinsically related to specific practices as they spontaneously arise and it is in this sense unique in character. Schon's concept of the reflective practitioner incorporates important aspects of systems development practices (Mathiassen 1998) and captures the essence of situated knowledge. For example, systems developers have to know the disciplines and methods of their profession. In addition however, they must have the capacity and skills to go beyond these forms of procedural knowledge and engage in inquiries into the problematic situation at hand. They need to engage in what Schon calls conversations with the situation by using analogies, diagnosis, and metaphors. In this way they reflect-in-action in situations involving complexity, uncertainty, instability, uniqueness, and value-conflicts. This form of knowing-in-action assumes that systems developers in their everyday life are knowledgeable in ways that are often tacit and implicit in their behaviour. On this basis they are able to apply procedural knowledge selectively, to combine elements from different methods, and even to invent new approaches as they go along.

KNOWLEDGE DYNAMICS

Based on our two-dimensional understanding of professional knowledge presented in Figure 1 a network of sixteen rather than four different knowledge conversions is potentially feasible. These are illustrated in Figure 2 (each arrowhead indicates a potentially relevant knowledge conversion). We suggest thinking of this complex network in a simplified way by focusing on conversions towards a particular form of professional knowledge. In the following we illustrate this by focusing on conversions towards situated knowledge and in doing so we relate to the ontological dimension of professional knowledge, i.e. the individual, the group, and the larger organisational network.

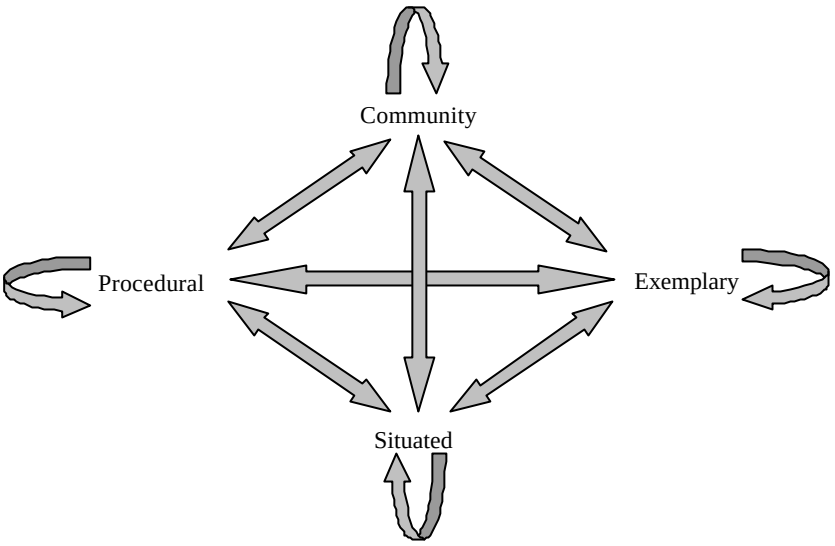

Figure 2. The network of potentially relevant knowledge conversions.

Conversions towards situated knowledge, i.e. knowledge creation activities where situated, community, procedural, and exemplary knowledge is converted into situated knowledge, are important in professional practices simply because situated knowledge by definition play a key role in producing results. Our framework suggests that there are four different ways in which systems developers can create or use knowledge in practical problem solving situations:

1. *Situated to situated.* Systems developers can use or reapply unique knowledge that has been created as a result of engaging in conversations with the problematic situation at hand. The situated knowledge is transferred from one situation to another through overlap between actors. The opportunity suggested by this form of conversion is to have the same systems developers participate in different activities, e.g. analysis, design, and programming, to facilitate conversion of their tacit and personal knowledge to other parts of a project. The risk here however is that a project replicates specific experiences without drawing upon state-of-the-art knowledge or insights from other similar projects.
2. *Community to situated.* Systems developers can enact the repertoire of assumptions, conventions, or traditions that is characteristic for the professional community they are part of. The opportunity suggested by this form of conversion is to transfer basic assumptions or conventions to a new project through careful selection of project members. If the members come from different communities-of-practice the project needs to focus on shared problem solving in order to confront diverse behavioural patterns and develop new community knowledge. The risk here however is that projects inherit inappropriate working habits or fail to develop a coherent set of shared practices.
3. *Procedural to situated.* System developers can adopt (aspects of) procedural knowledge that are available in the form of company specific methods and processes or in the form of state-of-the-art knowledge on systems development. The opportunity suggested by this form of conversion is to initiate change and learning based on the selective adoption of procedural knowledge. Implementation and further development of procedural knowledge is in this way viewed as an activity that is initiated and controlled by projects, rather than as a centralised activity. The risk here however is that projects uncritically adopt

procedural knowledge as espoused theories without appropriate concern for what it takes to actually change their theories-in-use as they are conditioned by community knowledge and enacted as situated knowledge.

4. *Exemplary to situated.* Systems developers can find inspiration in or reuse exemplary knowledge that is available in their organisational context. The opportunity presented by this type of conversion is for systems developers to reuse documents, models, and programs as part of each project, or for systems development organisations to facilitate cross-project learning through presentation and debate of relevant exemplary knowledge from innovative projects or experiments. The risk here however is that knowledge is uncritically transferred from one project to another without appropriate concern for the important contextual differences.

These conversions all target situated knowledge and they mostly relate to individuals and groups on the ontological level. Situated knowledge is embedded into the actual professional practices of systems developers and these are organised largely as a combination of individual and team activities. There are, however, also important interactions between the larger organisational network and the local activities in specific projects. Community, procedural, as well as exemplary knowledge can in various ways be transferred from the wider organisational network to individual and group practices in specific projects. Such conversions, are however, as indicated above not without severe risks.

We suggest that similar explorations make sense when one focuses on conversions targeting community, procedural, and exemplary knowledge. Such explorations reveal other opportunities and risks related to each specific type of conversion and they place different emphasis on the individual, group, and organisational levels. Reflecting, for example, on conversions towards procedural knowledge does imply a greater emphasis on the organisational level as compared to the reflections above.

IMPLICATIONS

We have argued that a more robust understanding of the relationships and differences between codified and explicit knowledge is fundamental to understanding and managing knowledge in contemporary organisations. This leads to an acknowledgement of four different types of professional knowledge and to a more elaborate understanding of knowledge conversion processes. On this basis we criticise conventional, one-dimensional conceptions of the epistemology of professional knowledge as overly simplistic and somewhat misleading. Nonaka & Takeuchi's framework (1995) focuses only on the differences between tacit and explicit knowledge and makes no distinction between explicit and codified knowledge. The weaknesses in this approach are that little attention is paid to (i) the problems and challenges involved in supporting knowledge management with information technology and (ii) the opportunities and risks related to codification of practices and procedures. Hansen *et al.*'s framework (1999) focuses on the differences between personalised and codified knowledge and makes no clear distinctions between tacit and personalised knowledge. The most important weakness here is that their framework pays no attention to codification of tacit knowledge within communities-of-practice. Moreover, the unintended implication of this perspective is that codified knowledge, being contrasted as it is with personalised or tacit knowledge, becomes *equivalent* to explicit knowledge, which we have clearly shown is not the case. Having highlighted and explored these important differences between codified and

explicit knowledge, there are a number of additional implications of our argument for knowledge management research and practice.

First, focusing on a particular domain of professional knowledge, say object-oriented modelling of user requirements, explicit knowledge is to some extent always codified. Explicit knowledge is at the very least codified on the level of syntax, but typically it will also apply codified forms of diagramming techniques (e.g. class diagrams and state-transition-diagrams) or language constructs (e.g. objects and classes). The issue is therefore not, whether knowledge within a specific domain is codified or not. It is rather a question of the degree to which knowledge is codified or the degree of tacitness (Boisot, 1995). In a similar fashion it hardly makes sense in practice to categorise knowledge as either being tacit or explicit. There will be certain aspects that are made explicit while other aspects remain tacit. We should consequently see the four types of knowledge as archetypes or perspectives that help us to understand the degree of explication and codification within particular areas of professional work.

Second, we need to rethink strategies for knowledge management. Hansen *et al.* highlight a choice between two main strategies - codification and personalisation - and suggest that an appropriate design will incorporate an 80%-20% mix for any given firm. Our framework suggests a much more complex picture with more options and variations. Activities and services vary within each professional work domain and there is a rich variety of potentially relevant knowledge conversions available (see Figure 2). While we share the view that organisations need to think critically about their approach to knowledge management, it hardly makes sense to simplify and generalise strategies as proposed by Hansen *et al.* A recommended approach within systems development organisations (Humphrey 1989) in order to improve the overall capability to develop computerised systems is to firstly initiate experiments within specific knowledge areas across a number of projects (situated knowledge). Then the experiences from each participating project should be explicated and evaluated (exemplary knowledge), and on that basis new processes should be developed (procedural knowledge) to be subsequently shared and institutionalised across projects (community knowledge). We can see from this very general example, a more complex dynamic occurring between different archetypes of knowledge than that proposed by Hansen *et al.'s* framework. On a more practical level, further elaboration is necessarily demanded as knowledge management practices are highly context dependent based on the particular type of professional activity in question (Mathiassen & Pourkomeylian 2002).

Finally it is necessary to rethink and elaborate the complex relationship between knowledge management and knowledge management systems. The use of information technology is increasingly seen as an important aspect of strategic management and many applications have recently been developed to support knowledge management practices within firms (Alavi & Leidner 2001). However, without a clear understanding of the differences and relationships between explicit and codified knowledge it is hardly possible to gain a practical understanding of the opportunities, limits, and risks related to the increased use of information technology for KM.

REFERENCES

Alavi, M. and D. Leidner, (2001). Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly*, Vol. 25, No. 1.

Applegate, L. M., F. W. Mcfarlan and J. L. Mckenney (1999). *Corporate Information Systems Management: Text and Cases*. McGraw-Hill.

Argyris, C. and D. Schon (1978). *Organizational Learning.* Addison-Wesley.

Basili, V. R. (1989). The Experience Factory: Packaging Software Experiences. Proceedings of the 14th Annual Software Engineering Workshop, NASA Goddard Space Flight Center, Greenbelt MD 20771.

Boisot, M. (1995). *Information Space: A Framework for Learning in Organizations, institutions and Culture*, Blackwell.

Brown, J. S. and P. Duguid (1991). Organizational Learning and Communities-of-Practice. Towards a Unified view of Working, Learning, and Innovation. *Organization Science*, Vol. 2, No. 1.

Brown, J. S. and P.Duguid (2001). Knowledge and Organization: A Social-Practice Perspective. *Organization Science*, Vol. 12, No.2.

Ciborra, C. and G. F. Lanzara (1994). Formative Contexts and Information Technology. Understanding the Dynamics of Innovation in Organizations. *Accounting, Management & Information Technology*, Vol. 4, No.2.

Coad, P. (1995). *Object Models. Strategies, Patterns, and Applications.* Yordon Press.

Dahlbom, B. and L. Mathiassen (1993). *Computers in Context. The Philosophy and Practice of Systems Design.* Blackwell.

Hansen, M. T., N. Nohria and T. Tierney (1999). What's Your Strategy for Managing Knowledge? *Harvard Business Review*, March-April.

Humphrey, W. (1989). *Managing the Software Process.* Addison-Wesley.

Jackobsen, I., G. Booch and J. Rumbaugh (1999). *The Unified Software Development Process.* Addison-Wesley.

Lyytinen, K. (1987). A Taxonomic Perspective of Information Systems Development. Theoretical Constructs and Recommendations. In R. J. Bolnd *et al.*: *Critical Issues in Information Systems Research.* John Wiley.

Mathiassen, L. (1998). Reflective Systems Development. *Scandinavian Journal of Information Systems,*Vol. 10, No. 1-2.

Mathiassen, L., J. Pries-Heje and O. Ngwenyama (2001): *Improving Software Organisations – From Principles to Practice.* Addison-Wesley.

Mathiassen, L. and P. Poukomeylian (2002). Knowledge Management in a Software Organisation. Submitted to *Journal of Knowledge Management.*

Mintzberg, H. (1983). *Structure in Fives. Designing Effective Organizations.* Prentice-Hall.

Nonaka, I. (1994). A Dynamic Theory of Knowledge Creation. *Organization Science*, Vol. 5.

Nonaka, I. & H. Takeuchi (1995). *The Knowledge Creating Company – How Japanese Companies Create the Dynamics of Innovation.* Oxford University Press.

Polanyi, M. (1962). *Personal Knowledge.* University of Chicago Press.

Polanyi, M. (1966). *The Tacit Dimension.* Anchor Day.

Schein, E. K. (1985). *Organizational Culture and Leadership. A Dynamic View.* Jossey-Bass.

Schon, D. (1983). *The Reflective Practitioner. How Professionals Think in Action.* Jossey-Bass.

Starbuck, W. (1992). Learning by Knowledge-intensive Firms, *Journal of Management Studies*, Vol. 29, No. 6.