Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2006 Proceedings

European Conference on Information Systems (ECIS)

2006

An exploratory study of an enterprise systems implementation methodology in action

D. Mihailescu
Jönköping International Business School, mida@ihh.hj.se

M. Mihailescu
Jönköping International Business School, mima@ihh.hj.se

Sven Carlsson
Lund University, sven.carlsson@ics.lu.se

AN EXPLORATORY STUDY OF AN ENTERPRISE SYSTEMS IMPLEMENTATION METHODOLOGY IN ACTION

Mihailescu, Daniela, Informatics, Jönköping International Business School, P.O. Box 1026, SE-551 11 Jönköping, Sweden, mida@ihh.hj.se

Mihailescu, Marius, Informatics, Jönköping International Business School, P.O. Box 1026, SE-551 11 Jönköping, Sweden, mima@ihh.hj.se

Carlsson, Sven A., Informatics, School of Economics and Management, Lund University, Ole Römers väg 6, SE-223 63 Lund, Sweden, sven.carlsson@ics.lu.se

Abstract

In this paper we develop an Enterprise Systems Implementation Methodology in Action framework. The framework is grounded in contemporary views on methodology in action (theoretical grounding) and enhanced by an empirical study (empirical grounding). The notion of Implementation Methodology in Action, as articulated in this paper, integrates two complementary views that conceptualise a methodology in terms of 1) a technology focusing on a formalised aspect, i.e. content as prescribed by designers, and 2) a structural view focusing on situational aspects including implementers, implementation context, Enterprise Systems Software and other individuals participating in the implementation project. The purpose of the framework is to integrate existing methodology views in order to contribute to a better understanding of Enterprise Systems implementation methodology. The integrated methodology concept is understood in terms of formalised and situational elements. These elements exert a significant influence on the characteristics and effectiveness of methodology use and implementers' actions, which may explain differences in the outcomes of the Enterprise Systems implementation process. The results from this research also helps participants in Enterprise Systems projects to become aware of the elements which may influence the actions of the users working with implementation methodologies as well as the reasons for these actions in the Enterprise Systems implementation.

Keywords: Implementation Methodology, Enterprise Systems, Methodology in Action, ERP.

1 INTRODUCTION

The Enterprise Systems (ES) market is one of the fastest growing markets in the software industry (Adam & Sammon 2004) and the largest part of a company's applications budget (approx. 34%) in 2002 (Somers & Nelson 2004). ES are regarded as a significant break from past practice for the following reasons: 1) ES have a comprehensive scope, 2) they are based on vendor-developed application software, and 3) they include a mix of business process change in organisation and software configuration and/or customisation to align the software with the business processes. When implementing ES, most organisations are required to do significant changes and to (re)design processes according to the 'best practices' embedded in ES. This is considered the most challenging issue in ES projects (Davenport 2000, Shanks et al. 2003). Robey et al. (2002) suggest that ES implementation can be understood as a dialectic of learning. Their findings indicate that firms have to overcome knowledge barriers associated with the configuration of the Enterprise Systems Software (ESS) and with the assimilation of new work processes.

ES create new opportunities but also new challenges and difficulties for both users and implementers and influence ES development. The ES development consists of two life cycles: 1) the development of generic ESS, e.g. ES application software by vendors, like SAP, and 2) the ES implementation life cycle that takes place in a context, i.e. at a user organisation, and includes the implementation of ESS (Hedman 2003). The clear distinction between the development and implementation process of ESS seems to influence not only the content and characteristics of methodologies, i.e. underlying assumptions, phases, activities and related methods, techniques and tools prescribed by vendors but also their roles and utilisation, i.e. to support the implementation and not the development of ESS.

Implementation methodologies used in implementation processes of ESS in organisations affect the outcome of the often complex and problematic implementation processes (Esteves & Pastor 2001, Hedman 2003, Somers & Nelson 2004). Although most ESS incorporates similar functionality, most ESS vendors have developed their own implementation methodologies. ES implementation methodologies are emerging in practice, but studies regarding ES implementation methodologies are still rare. Esteves and Pastor (2001) argue that there is a lack of studies focusing on the utilisation of ES methodologies in an organisational context. We address this lacuna by exploring the characteristics of an ES implementation methodology developed by an ESS vendor.

The rest of the paper is organised as follows. Next the basis for our work and the conceptual framework is presented. This is the theoretical grounding of our framework; it is followed by our study of an ES implementation methodology which represents the empirical grounding of our framework. Our findings are discussed in the final section along with implications for theory and practice.

2 METHOD IN ACTION

Information Systems Development (ISD) methodology research tends to focus on the formalised aspect of the methodologies (*technology*) or considers the use of the methodologies in practice (*structure*). Methodology as *technology* focuses on the *formalised aspect*, i.e. the content of a methodology as prescribed by methodology designers and perceived by implementers (methodology users). Methodology as *structure* focuses on the *situational aspect*, i.e. how implementers enact a methodology in practice. An overview of the main issues concerning methodologies and their use, based on both theoretical and empirical findings, is outlined in the *Method in Action* (MiA) framework suggested by Fitzgerald et al. (2002). In this framework it is suggested that method in action is *shaped* by the development context and *influenced* by the rational and political roles which *justify* a formalised method which *may be the basis* for it. Furthermore, it is considered that the information processing system is *developed* by *analysing* the development context and *enacting* the method in action for each distinctive situation.

The framework focuses on the following interdependent elements: *development context*, *developers*, *information processing system*, *formalised method*, *roles of method* and *method in action* (Fitzgerald et al. 2002). Although the authors acknowledge that developers have a central role in the development process, this is not elaborated on or reflected in the framework. These aspects are highlighted by Mathiassen (1997) who applies Schön's *reflection in action* with regard to developers, and by Sambamurthy and Kirsch (2000) who suggest an *ISD process* framework. Sambamurthy and Kirsch (2000) suggest that developers engage in transactions and use methodology and other structures in order to justify transactions, i.e. developers judge outcomes of these transactions and act accordingly. This point of view is expressed also by Mathiassen (1997) who suggests that developers reflect on the situations in which they find themselves, i.e. reflection in action, and from there adapt a combination of formal and informal approaches. The frameworks are complementary in offering a picture of the use of methodology on an individual level, i.e. individuals' actions in relation to the ISD process.

Fitzgerald et al.'s (2002) MiA framework underpins our research. The framework, with its elements, is adapted to explore the nature of ES implementation methodologies. The elements considered are: *formalised implementation methodology and its roles*; *implementation context*; *implementers*; *ESS* and *implementation methodology in action*. The elements are grouped in terms of formalised and situational aspects (Table 1).

	Formalised aspect	Situational aspect
Conceptualisation of Methodology	Methodology as a Technology	Methodology as a structure
Element	Formalised Implementation Methodology and its roles	Implementers Implementation Context Enterprise Systems Software
Integrated Methodology conceptualisation	Implementation Methodology in Action	

Table 1. Conceptual framework based on the Method in Action framework (Fitzgerald et al. 2002).

Building on the work of Fitzgerald et al. (2002), Mathiassen (1997) and Sambamurthy and Kirsch (2000), our aim is to develop a more comprehensive framework that takes into account formalised and situational aspects of an implementation methodology and captures both implementers' actions and the elements that influence them in relation to ES implementation. The framework combines elements from existing theoretical frameworks (theoretical grounding) and, based on an empirical study, we enhance the framework (empirical grounding). By doing this, we provide an extended and systematic way to organise implementation methodology considerations in relation to ES implementation.

3 RESEARCH APROACH

In this study a qualitative case research strategy is used, which enables us to explore and understand the formalised and situational aspects of a contemporary ES implementation methodology. The case, SAP's Accelerated SAP (ASAP), was selected based on two case sampling strategies, i.e. critical and typical sampling. The implementation methodologies proposed for ES projects seem to be similar (Esteves & Pastor 2001, Fitzgerald et al. 2002). Consequently, it is likely that the implementation methodology which is developed and proposed to be used in ES implementation projects by SAP, the leading Enterprise Resource Planning (ERP) software provider, illustrates typical characteristics of implementation methodologies for ES projects.

The case study may be regarded as a critical case since the methodology studied is used by consultants in implementations all over the world, is well documented and has been presented at conferences and in research publications by, among others, Smethurst and Kawalek (1999), Ghosh (1999), van Slooten and Yap (1999), Welti (1999), Hedman (2003), and Esteves et al. (2003). This facilitates finding sources of data that are information rich and permit logical generalisation.

The MiA framework (Fitzgerald et al. 2002) is used to structure the study as well as to generate questions to be asked in interviews and to provide a guide to deciding what documents to study. The purpose of the empirical data analysis is to enhance our framework.

The primary data collection method was semi-structured interviews to elicit the views and experiences of ten ES implementers. The respondents were selected by applying a chain sampling strategy. Five implementers from consultancy firms and SAP consultants who were familiar with the ASAP implementation methodology were interviewed. An additional five interviews were conducted at SAP's conference "Managing SAP Projects 2004". The interviews were done in the autumn of 2004; each interview lasted between 30 minutes and two hours. Notes were taken during all interviews and some of the interviews were tape-recorded and then transcribed. Data were analysed after each interview. In this way it was possible to learn from one interview to the next.

Documentation was the second data source. Documents provided by some interviewees, SAP's web pages, the SAP conference and published articles and books were used. Data for addressing the formalised aspect of ASAP implementation methodology came mainly from documents; for addressing the situational aspect primarily interview data were used. To increase the quality of our study we intentionally tried to follow Yin's (2003) recommendation on using data triangulation.

4 ASAP IN ACTION

SAP AG, the largest ERP provider, has developed and recommends the use of ASAP, which represents the de facto standard for implementing SAP solutions. The ASAP methodology was released in June 1997 and in 2003 it was integrated in a tool named Solution Manager and in all SAP installations without charge (SAP consultant, www.sap.com 28 February 2005). ASAP has been continuously improved and updated by an international consulting team which collects feedback from SAP customers using ASAP (SAP consultant from Spring Consulting Sweden).

Our presentation and discussion of ASAP in Action is structured according to the two aspects and their elements presented in Section 2 (see Table 1).

4.1 Formalised aspect of ASAP implementation methodology

The aim of ASAP implementation methodology is to facilitate and support a structured, efficient and accelerated implementation of SAP's solutions during the entire implementation life cycle, i.e. *Discovery, Evaluation, Implementation* and *Operation*. The majority of SAP partners customise, i.e. add, adapt or supplement parts to ASAP and develop their own methodology called Powered ASAP. The ASAP implementation methodology specifies necessary project management activities and development sequences as well as methods, techniques, tools and services to be used in order to structure and support the entire implementation phase. Based on our empirical data our summary of formalised characteristics of ASAP is presented in Table 2 below.

Concept	ASAP implementation methodology
Approach	*Continuous Business Engineering* – business process innovation through technology and IT practice and rapid implementation of flexible and market-based organisation structures in enterprises with related data processing support • *goals*: standardise, structure and guide a rapid and efficient implementation of SAP solutions in organisations; • *assumptions*: an enterprise should accept the embedded 'best practices' wherever possible; a common frame of reference, i.e. a business blueprint, is created by using a Reference Model which incorporates standardised and competition-neutral process flows and can be used as a starting point to solution and adjusted to a specific situation in that way avoiding having to begin the modelling effort from scratch and instead profit from the experiences of other enterprises; • *process model*: linear sequence of phases for entire implementation life cycle: Discovery, Evaluation, Implementation and Operation; • *conceptual model*: process oriented.
Methods	*ASAP Roadmaps* including activities, deliverables, role descriptions, additional guides and accelerators as well as associated techniques and implementation tools; *Iterative Process Prototyping* with a focus on process design based on Reference Model and Prototyping.
Techniques	*Event-controlled Process Chain* represents the basis for all process modelling activities and structures and ensures consistent specification of process requirement, i.e. describes process flow of events, tasks and processes.
Tools	*Accelerators* and comprehensive *application platforms* for structuring and supporting the construction and maintenance of the business blueprint and implementation phases; Examples: SAP Solution Composer, ASAP Roadmaps, SAP Solution Manager.
Services	Accessed through the operation or triggered dynamically depending on the configuration status of a solution.

Table 2. Formalised characteristics of ASAP implementation methodology.

ASAP can be customised to fit a specific project scope and a type of solution by choosing different roadmaps with related activities and accelerators which are dynamically adapted by using integrated implementation tools. Additionally these implementation tools are integrated with the application software permitting (ideally) rapid implementation and optimisation by providing guidelines and so-called accelerators, i.e. templates and tools, in order to manage the implementation process, configure/customise the ESS, in this case SAP's enterprise systems software *mySAP*, and produce related documentation (Ghosh 1999). Since the content of the tools is dynamically adapted and integrated with the application software, the implementation tools are regarded as one of the key components of ASAP. The underlying rational assumptions of ASAP were noticeable among *roles* found in the empirical material.

The roles are summarised in Table 3. The first three represent typical examples of rational roles while the fourth and fifth roles can be regarded as political roles. However, the last three roles draw the attention to understanding, knowledge and communication, which are related to the enactment of a methodology in an actual project, i.e. the use of ASAP, and highlight the situational aspect of ASAP implementation methodology.

	Roles
1	To have a transparent approach which helps to integrate, harmonise, coordinate, structure, control, guide and follow up components that make an implementation successful, i.e. a rapid delivery of the system and working as a team to reduce implementation time, costs and risks.
2	To describe all activities in an implementation and make sure that nothing is left out by offering project plans well in advance for execution and ensuring the quality of the work by being an integrated part of the quality assurance system.
3	To include the entire technical area and support technical project management by addressing things like interfaces, data conversion and authorisation earlier than in most traditional implementations.
4	To achieve better results by using accelerators and best practices which cover both hard facts, i.e. what to do, as well as soft facts, i.e. who, when and how to do things.
5	To help implementers to understand the context, facilitate their communication with customers and the work with an organisation in change and handle unexpected changes in a project.
6	To win credibility and trustworthiness and reduce resistance.
7	To meet requirements in order to be certified by vendors and accepted customers.
8	To promote knowledge management, e.g. to gather knowledge on how to organise and run implementation projects and document the knowledge.

Table 3. Roles of ASAP implementation methodology.

4.2 Situational aspect of ASAP implementation methodology

The elements which reveal the situational aspect of a methodology are: *implementers*, *implementation context*, and *Enterprise Systems Software* (in this case SAP's mySAP).

4.2.1 Implementers

Implementers have a decisive impact on the situational aspect of a methodology since they uniquely enact a methodology (Fitzgerald et al. 2002) in order to structure their work, understand the context, communicate with customers and other implementers as well as configure/customise the ESS. The formalised content of a methodology, i.e. accelerators and checkpoints, may help implementers to reflect on and judge the outcomes of different activities (Mathiassen 1997, Sambamurthy & Kirsch 2000). To act in a specific context and situation the implementers use their competence and if necessary adapt the formalised content of the methodology. Our study shed some light on what competences are found necessary for implementers.

General business knowledge and competences include organisational, strategic, business process and project management. Esteves et al. (2003) argue that these have the highest relevance in SAP implementation. The empirical data suggest that the recruitment of implementers is based not only on their technical knowledge of SAP products such as integration, configuration, special modules or SAP's ABAP programming language but in many cases on their business competences. The data also suggest that business competences are more difficult to acquire since this type of competence to a large extent is based on skills and experience. One aim of recruiting implementers is to reduce the gap between business and technical competences (Crysalis Consulting Services consultant). Besides business and technical competences the third category of competences, i.e. personal competences (Kræmmergaard & Rose 2002), appears to receive increased attention and the literature suggests that it has considerable importance in ES projects (Palaniswamy & Frank 2000, Scott & Kaindl 2000).

A summary of personal competences perceived as important by interviewees is presented in Table 4.

	Personal competences
1	Commitment, motivation, inquisitiveness in order to ask relevant questions, collect information and understand the context.
2	Creativity, experience and professional knowledge, e.g. business and technical, necessary to suggest the right solution.
3	Open attitude, professionalism and ability to communicate the solution in order to gain trust.
4	Communication and collaboration skills and ability to work in teams and manage conflicts.

Table 4. Perceived personal competences by implementers.

The first three are intra-personal competences necessary to understand the context, suggest solutions and gain trust, while the last competence reflects essential inter-personal competences required to work in project teams. Our data suggest that general business competences will remain significant, but an increased specialisation of competences may accentuate the importance of personal competences and especially communication and collaboration skills. This gradual movement, called *vertical specialisation* or branch specialisation, was especially noted in our US data. The vertical specialisation of competences on an organisation level seems to have strategic or political motivation, e.g. the quality of a solution and competitiveness, and is regarded as necessary in order to deal with the complexity which characterises modern ES solutions and ESS.

4.2.2 Implementation context

The context has different meanings in relation to implementation methodology. The context has a direct and an indirect influence. A direct influence on the implementers' decisions and work with methodology (Fitzgerald et al. 2002, Mathiassen 1997, Sambamurthy & Kirsch 2000) is represented by the contextual characteristics, uniqueness, complexity and uncertainty, and is reflected on project and organisation levels. The characteristics of a project, i.e. scope and goal of a project along with the implementation approach, are considered significant to elucidate expectations and perceptions of project participants, both implementers and customers, related to

- the degree of organisational change, and
- the extend to which the ESS needs to be configured/customised.

The characteristics of an organisation mentioned by interviewees are summarised in Table 5 and are complementary to project characteristics. Thus, the project and organisation characteristics are on the one hand considered to help implementers to understand and recognise the characteristics of the context and on the other hand to influence the roadmap, resources and time necessary to accomplish the tasks during the implementation.

	Characteristics of an organisation
1	Culture and structure
2	Business values and specific strategic aspects
3	Degree of standardised processes and use of standardised components
4	Customers' understanding of the organisation
5	Technological infrastructure

Table 5. Characteristics of an organisation.

The interviewees also mentioned two other contextual characteristics that seem to influence the developers' use of methodology, albeit in an indirect way:

- a developer's context which has a direct influence on implementers and represents the context where the methodology is adopted and may be adapted,
- a designer's context which has a direct influence on formalised characteristics of the methodology.

These two contexts are less focused on in the ISD literature but are more emphasised in innovation literature. Hence, a combination of the two literature streams is considered to be beneficial and relevant for a better understanding of the nature of a methodology.

4.2.3 Enterprise System Software

According to SAP, the purpose of mySAP software is to make use of technological advances and support the business requirements of an adaptable business. It automates and supports the business process but most importantly, it enables organisational changes based on best practices. The current business solutions offered by SAP comprise three parts:

- a highly integrated application platform, *SAP NetWeaver,* which includes implementation tools like *SAP Solution Manager*,
- a well defined and tuned business core functionality existent in business application software, *mySAP ERP,*
- additional enterprise extensions to support collaborative processes, *mySAP Solutions*.

4.3 ASAP implementation methodology in action

In order to take full advantage of the underlying approach, adherence to integrated implementation tools which automatically adapt the content of existing roadmaps in accordance with the characteristics of a project and type of solution, is recommended. In this way the actual implementation process emerges in compliance with a stipulated implementation process and the adaptation of methodology is reduced. However, implementers have the possibility to influence the content of activities and individualise the accelerators, such as templates, related to each phase. Through formal or informal transactions with other project participants they understand the context by using a Reference Model as baseline. Consequently, the *enactment* of ASAP implementation methodology in practice is influenced explicitly or implicitly by a *formalised implementation methodology*; *other individuals' agendas* or *expectations*; *ESS* which includes integrated implementation tools, e.g. what implementation methodology is used, and is shaped by the *implementation context*. This implies that an implementer *reflects* on formal activities and transactions (specific informal and formal means through which stakeholders achieve their agendas or ensure that appropriate tasks are completed) or those which occur spontaneously in their work and enact the implementation method in action, i.e. adapt the use of methodology and take action, with regard to the characteristics of mentioned elements in order to

- *manage* and *document* the implementation process over time, e.g. phases, activities, methods and techniques with defined milestones and deliveries,
- *identify, design* and *model* a business blueprint based on the Reference Model embedded in mySAP with regard to regulations and characteristics of the implementation context,
- *configure and/or customise* mySAP, e.g. parameter settings and/or addition of extra features to the software by writing program code, modules and/or system integration,
- *update* the formalised methodology over time in terms of phases, activities, templates, methods, techniques and tools, based on implementers' own experience and/or regulations from their organisation,
- *communicate*, *collaborate* and *negotiate* with managers, users and other implementers involved in a project.

The enactment of ASAP implementation methodology is illustrated in Figure 1. It will be discussed in the next section, in terms of elements which influence (continuous arrows in the figure) the use of ASAP implementation methodology and implementers' actions (dashed arrows in the figure) which occur through the use of implementation methodology in the ESS implementation process.

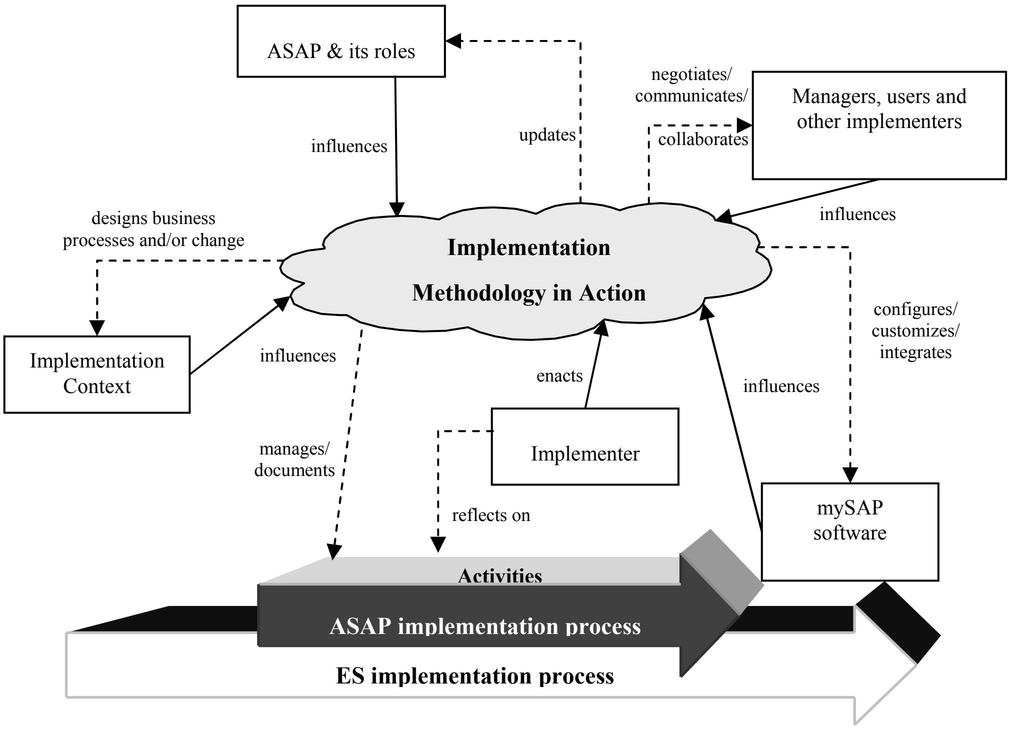

Figure 1. ASAP Implementation Methodology in Action.

5 DISCUSSION

The ASAP study illustrates the notion of *implementation methodology in action*, i.e. enactment, as articulated in the study, and draws the attention to the fact that an underlying approach permeates the content of implementation methodology which is explicitly or implicitly enacted by implementers reflecting on transactions/activities with regard to

- other individuals' agendas and expectations; ESS; and implementation context,

in order to:

- design business processes and/or change the implementation context; manage and document the implementation process; configure/customise the ESS; update the formalised implementation methodology with regard to recommendations from their own organisation and/or earlier projects; negotiate, communicate and collaborate with other individuals participating in the implementation project.

Although ASAP in many cases is requested by customers there are organisations which use their own methodology or other non-ASAP methodology. According to all interviewees the different implementation methodologies on the market are similar to ASAP. The main difference between them is considered to be 1) the number of accelerators which are significant in order to identify when and what to do, 2) the integrated tools and prototyping which support and have a significant importance during the implementation process, and 3) some fixed principles specific to SAP's application software that need to be taken into consideration.

In the case of SAP the implementation tools to design and model processes as well as to configure and optimise the application software are integrated within a comprehensive application platform, SAP NetWeaver. Furthermore the tool for configuration and optimisation, i.e. SAP Solution Manager, is integrated with the application software in all SAP installations without charge. This movement towards non-fee based accessibility and ownership of integrated implementation tools with the application software, may lead to increased interest and probably a request for the utilisation of implementation tools and consequently an increased use of the ASAP implementation methodology.

The formalised aspect of ASAP implementation methodology is in general consistent with the terminology and observations in the ISD literature. For instance, the methodology is constructed of standardised method fragments from existing methods in concordance with the characteristics of ES projects by integrating standardised method fragments from both management and development and additional techniques and tools. Hence, ASAP implementation methodology represents a framework which is adapted from the construction and is designed to support and accelerate the implementation of SAP's solutions. However, even if the methodology is adapted during its construction as suggested by Brinkkemper et al. (1999), this adaptation, as well as the one which may occur in implementers' organisations when the methodology is adopted, is related to a methodology's formalised aspect. For instance, the purpose of a methodology is limited to well-defined project types but not to different choices that implementers have to make when they use a methodology with regard to different organisational situations. Therefore a dynamic adaptation based on the characteristics of a particular project, type of solution and organisational context as well as formal and informal transactions occurs during the implementation process when an implementer individualises activities, templates and accelerators and uses a methodology in practice, i.e. situational aspect. Thus, a methodology is characterised by formalised and situational aspects and adapted on organisation, project and individual level, which may explain differences in the outcomes of projects.

Paradoxically, the information-predetermined and bureaucratic character of ASAP methodology (Hedman 2003) along with requirements of changes in organisations and utilisation of comprehensive implementation tools causes dissatisfaction (Truex & Avison 2003). However, at the same time these features are a prerequisite for managing and succeeding with the implementation (van Slooten & Yap 1999). Additionally the incorporation of implementation methodology into tools integrated with the application software has the effect of structuring and standardising the implementation process by enforcing a particular set of phases, activities and necessary documentation on implementers (Truex & Avison 2003). However, in this way the implementer is guided through the configuration process of the application software. This may explain interviewers' views that the configuration is less arduous than the design of solution, i.e. identification and modelling processes, views that support the results presented by Esteves et al. (2003) who suggest that business knowledge represents the most significant knowledge during the entire implementation process. Furthermore, since implementation tools are owned by the customers who have experience and knowledge about their own organisation, implementers' job descriptions seem to change and may be comparable to a coach activity. Possible change of job descriptions has been indicated in the literature by Scott and Kaindl (2000), who suggest that consultants facilitate open and productive communication. This is supported by the empirical material in which an implementer's work is compared to the work of an architect who collects information and builds a comprehensive image and solution. For implementers a methodology shows how to reach the solution and represents an instrument to communicate and help an organisation to understand the complexity of the environment and the solution, to change behaviour and mature through a transparent process during the implementation (SAP consultant from Spring Consulting Sweden). ASAP implementation methodology like other formalised methodologies provides support for explicit knowledge (related to formalised aspect, i.e. the content of a methodology). Templates and tools may be regarded as knowledge repositories for organisations and implementers who reuse and update templates with knowledge from earlier projects. However, the use of a methodology and the ability to deal with unexpected events is more a question of tacit knowledge which is related to implementers' competences, i.e. skills and abilities such as creativity, motivation and inquisitiveness, which cannot be easily planned and learned.

6 CONCLUSION

This paper has proposed a framework which draws on contemporary views on methodology in action and is enhanced by a case study for integrating issues around formalised and situational aspects of implementation methodologies. We argue that implementers reflect on activities and enact implementation methodology in order to

- *design* business processes and/or change the implementation context,
- *manage* and *document* the implementation process,
- *configure/customise* the ESS,
- *update* the formalised implementation methodology with regard to recommendations from their own organisation and/or earlier projects, and
- *negotiate*, *communicate* and *collaborate* with other individuals participating in the implementation project.

The enactment of the implementation methodology in action is influenced by a formalised implementation methodology, other individuals' agendas and expectations, ESS and the implementation context. In this way an implementation methodology is understood in terms of *elements* which exert a significant influence on the characteristics and effectiveness of methodology use, and implementers' *actions,* which may explain differences in the outcomes of the ES implementation process.

With regard to the formalised characteristics of implementation methodology employed in ES implementation, it can be concluded that the essential characteristic expressed in the underlying approach in terms of philosophy, assumptions, goals, process and conceptual model, along with an accentuation of integrated implementation tools and other accelerators offered to implementers as well as users, represents a new, or in any case, a different approach from existing ISD approaches. The integration of formalised and situational aspects is considered to highlight the characteristics of the implementation methodology concept and is regarded as necessary in order to increase our understanding of this concept and reduce possible misunderstandings and confusing research results. Since the Continuous Business Engineering approach is relatively new and promulgates the principle of accelerated implementation based on a Reference Model and continuous business process innovation, this approach may cause *misunderstandings, faulty expectations* and *culture clash* between implementers and customers. Such problems may occur if the focus is on the use of methodology only, without consideration and communication of its underlying approach, i.e. philosophy, basic assumption and goals. Additionally, consideration only of a methodology's formalised aspect without regard to the implications of its utilisation leads to different and in part confusing research results. We believe that our conceptualisation may be of value to researchers investigating similar phenomena in other settings or ES implementations and implementation methodology enactment by providing concepts for understanding an implementation methodology.

Further empirical investigations are needed in order to provide more evidence and improve our understanding of ES implementation methodologies. Through intensive studies of the implementation process of SAP's solutions and other ESS vendors' solutions in practice, the components and relationships included in the framework can be explored further in more detail in order to provide more solid evidence for how the implementation methodology is actually used and to further disclose the differences in use of implementation methodologies. In order to improve the generalisability of the findings the framework can be applied to other implementation methodologies on the market and address cross-cultural aspects in order to uncover global and regional issues related to the use of ES implementation methodologies. The implementation methodology is relevant not only to implementers but also to users, and therefore it could be interesting to focus on implementation methodology support for improvement and process optimisation as well as necessary competence to handle business processes in an innovative way to sustain growth and competitiveness of organisations.

References

Adam, F. and D. Sammon (Eds.) (2004). The Enterprise Resource Planning Decade: Lessons Learned and Issues for the Future. Idea Group Publishing, Hershey, PA.

Brinkkemper, S., Saeki, M. and F. Harmsen (1999). Meta-modelling based assembly techniques for situational method engineering. Information Systems, 24 (3), 209-228.

Davenport, T.H. (2000). Mission Critical: Realizing the Promise of Enterprise Systems. Harvard Business School Press, Boston.

Esteves, J., Chan, R., Pastor, J. and M. Rosemann (2003). An exploratory study of knowledge types relevance along enterprise systems implementation phases. The Fourth European Conference on Organizational Knowledge, Learning and Capabilities, Barcelona.

Esteves, J. and J. Pastor (2001). Enterprise resource planning systems research: An annotated bibliography. Communication of the AIS, 7 (8), 1-52.

Fitzgerald, B., Russo, N.L. and E. Stolterman (2002). Information Systems Development: Methods in Action. McGraw-Hill, London.

Hedman, J. (2003). On Enterprise Systems Artifacts: Changes in Information Systems Development and Evaluation. PhD Dissertation. Department of Informatics, Lund University.

Ghosh, J. (1999). SAP Project Management. McGraw-Hill, London.

Kræmmergaard, P. and J. Rose (2002). Managerial competences for ERP journeys. Information Systems Frontiers, 4 (2), 199-211.

Mathiassen, L. (1997). Reflective Systems Development. Department of Computer Science, Aalborg, Denmark: (http://www.cs.auc.dk/~larsm/rsd.html - accessed 10 October 2004).

Palaniswamy, R. and T. Frank (2000). Enhancing manufacturing performance with ERP systems. Information Systems Management, 17 (3), 43-55.

Robey, D., Ross, J.W. and M.-C. Boudreau (2002). Learning to implement enterprise systems: An exploratory study of the dialectics of change. Journal of Management Information Systems, 19 (1), 17-46.

Sambamurthy, V. and J.L. Kirsch (2000). An integrative framework of the information systems development process. Decision Sciences, 31 (2), 391-412.

Scott, J. and L. Kaindl (2000). Enhancing functionality in an enterprise software package. Information & Management, 37 (3), 111-122.

Seddon, P.B., Shanks, G. and L.P. Willcocks (2003). Introduction: ERP – The quiet revolution? In Second-Wave Enterprise Resource Planning Systems: Implementing for Effectiveness (Shanks, G., Seddon, P.B. and L.P. Willcocks (Eds.), Cambridge University Press, Cambridge.

Slooten, K. van, and L. Yap (1999). Implementing ERP information systems using SAP. In Proceedings of the Fifth Americas Conference on Information Systems (AMICS), Milwaukee, 226-228.

Smethurst, J. and P. Kawalek (1999). Structured methodology usage in ERP implementation projects: An empirical investigation. In Proceedings of the Fifth Americas Conference on Information Systems (AMICS), Milwaukee, 219-221.

Somers, T.M. and K.G. Nelson (2004). A taxonomy of players and activities across the ERP project life cycle. Information & Management, 41 (3), 257-278.

Truex, D. and D. Avison (2003). Method engineering: Reflections on the past and ways forward. In Proceedings of the Ninth Americas Conference on Information Systems (AMICIS), Tampa, Florida.

Welti, N. (1999). Successful SAP R/3 Implementation: Practical Management of ERP Projects. Addison Wesley Professional, UK.

Yin, R.K. (2003). Case Study Research: Design and Methods. 3rd Edition. Sage Publications Thousand Oaks, CA.