JYVÄSKYLÄ STUDIES IN COMPUTING
150

Irja Kankaanpää

IT Artefact Renewal

Triggers, Timing and Benefits

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 150

Irja Kankaanpää

IT Artefact Renewal

Triggers, Timing and Benefits

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen salissa AgAud 2
joulukuun 20. päivänä 2011 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, auditorium AgAud 2, on December 20, 2011 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2011

IT Artefact Renewal

Triggers, Timing and Benefits

JYVÄSKYLÄ STUDIES IN COMPUTING 150

Irja Kankaanpää

IT Artefact Renewal

Triggers, Timing and Benefits

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ 2011

Editors
Seppo Puuronen
Department of Computer Science and Information Systems, University of Jyväskylä
Pekka Olsbo, Harri Hirvi
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-4602-9
ISBN 978-951-39-4602-9 (PDF)

ISBN 978-951-39-4601-2 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2011

ABSTRACT

Kankaanpää, Irja
IT Artefact Renewal: Triggers, Timing and Benefits
Jyväskylä: University of Jyväskylä, 2011, 79 p. (+ included articles)
(Jyväskylä Studies of Computing,
ISSN 1456-5390; 150)
ISBN 978-951-39-4601-2 (nid.)
ISBN 978-951-39-4602-9 (PDF)
Finnish summary
Diss.

This thesis studies the life cycle of information systems (IS) from an organizational perspective. It states that the life cycle of an IS, as a socio-technical system, is cyclical rather than linear. Within this cyclic view, the IS is renewed at certain intervals in order to stay aligned with organizational needs. An information system is a sum of social and technical elements. Misalignment between the elements creates pressure for change. When one of the elements changes the others adapt. The renewal of the technical element of an IS, i.e. the IT artefact, is labour-intensive and expensive. Consequently, the related decisions have to be well justified. Renewal triggers provide indication of the motive and criticality of the need for change. Benefit evaluation aims at ensuring that the renewal effort yields benefits and is worth the investment of organizational resources. Correct timing tackles with finding a time period during which the IT artefact renewal causes minimum hindrance for the organization's operation, and defining the suitable frequency of renewal projects. Together these three aspects determine the IT artefact renewal.

Keywords: Information system, IT artefact, life cycle, IT management

Author's address Irja Kankaanpää
University of Jyväskylä
Dept. of Computer Science and Information Systems
P.O. Box 35 (Agora)
40014 University of Jyväskylä, Finland
irja.k.kankaanpaa@jyu.fi

Supervisors Professor Samuli Pekkola
Institute of Business Information Management
Tampere University of Technology

Professor Pasi Tyrväinen
Dept. of Computer Science and Information Systems
University of Jyväskylä

Professor Jarmo J. Ahonen
Department of Computer Science
University of Kuopio

Reviewers Professor Mike Newman
Manchester Business School
The University of Manchester, UK

Professor Steve Sawyer
School of Information Studies
Syracuse University, U.S.A

Opponent Professor Matti Rossi
School of Economics
Aalto University, Finland

ACKNOWLEDGEMENTS

I want to give my deepest thanks to professor Samuli Pekkola, professor Pasi Tyrväinen and professon Jarmo J. Ahonen who have directed this thesis. I am grateful to Janne Lepistö whose words gave motivation and inspiration for the thesis in the first place. I want to express my gratitude to COMAS, Liikesivistysrahasto and the rector of University of Jyväskylä for the financial support they granted for the work with the thesis. I also want to thank David Agar for his patience and support in navigating through the challenges that emerged during the research process. I want to thank Tero Tilus, Henna Sivula Lahtonen, Päivi Juutilainen, Jussi Koskinen and Heikki Lintinen for their comments and joint work. I also want to thank Petri Maaranen for his help and ideas during the research process. I want to thank the coffee room people of information systems and computer science department, particularly Maritta Pirhonen, Liisa Kuparinen and Mirja Pulkkinen for the discussions and support along the process of writing the introduction of the thesis.

In Jyväskylä 24.11.2011
Irja Kankaanpää

FIGURES

TABLES

LIST OF INCLUDED ARTICLES

1. Kankaanpää, I. and Pekkola, S. (2011) The life of information systems. Manuscript, University of Jyväskylä. Submitted to Journal of Information Technology Theory and Application (JITTA).

2. Kankaanpää, I. (2010) Information system's life cycle: Interpretative study on CIO perceptions. Submitted to Scandinavian Journal of Information Systems (SJIS) 6.7.2011.

3. Kankaanpää I., Tiihonen P., Ahonen J. J., Koskinen J., Tilus T., and Sivula H. (2007) Legacy System Evolution – A Comparative Study of Modernisation and Replacement Initiation Factors. In A. Costa, V. Duarte, B. Encarnação, L. Marques and V. Pedrosa (Eds.) the Proceedings of the Ninth International Conference on Enterprise Information Systems (ICEIS 2007), DISI (pp. 280-287), DISI, Madeira: INSTICC.

4. Kankaanpää I. and Pekkola S. (2010) Timing the information system upgrade. In P. M. Alexander, M. Turpin and J. P. van Deventer (Eds.) the proceedings of the 18th European Conference on Information Systems (ECIS2010) (pp. 1545-1557). Pretoria: University of Pretoria.

5. Kankaanpää I., Koskinen J., Tilus T., Sivula H., Lintinen H. and Ahonen J. J. (2007) IS Evolution Benefit Assessment - Challenges with Economic Investment Criteria. In W. Abramowicz and H. C. Mayr (Eds.) Technologies for Business Information Systems (pp. 183-191). Netherlands: Springer.

6. Kankaanpää I., Sivula H., Ahonen J. J., Tilus T., Koskinen J. and Juutilainen P. (2005) ISEBA – A Framework for IS Evolution Benefit Assessment. In D. Remenyi (Ed.) the Proceedings of the 12th European Conference on Information Technology Evaluation (ECITE 2005) (pp. 255-264), UK: Academic Conferences.

CONTENTS

1 INTRODUCTION

The life cycle metaphor has been used in information systems (IS) field to describe the development stages of information systems (ISD) (Avison and Fitzgerald 2008) and the software development life cycle (SDLC) (Kazman, Nord and Klein 2003, Sen and Zheng 2007, Davis, Bersoff and Comer 1988). However, these development models are inadequate for capturing the IS life cycle from organizational point of view (Sawyer 2001) as they fail to engage organizational change with the software change (McGann and Lyytinen 2008). This is also the problem with acquisition centred models even though they incorporate organization's perspective (see e.g. Esteves and Pastor 1999, Markus and Tanis 2000, Chang, Yen, Huang and Hung 2008) but ignore the iterative nature of change in IS life cycle (Lyytinen and Newman 2008, p. 289):

> "A majority of change studies treat the change as a simple, linear progression where a new (technical) system is designed, adopted, and modified in step-wise manner."

Consequently, there is a need within IS discipline for wider perspective on IS change that engages software change with organizational change (McGann and Lyytinen 2008).

A more holistic approach can be found in the work system life cycle (WSLC) model (Alter 2001). The work system is a general representation of a system that produces goods or services, e.g. a system that produces horse shoes. Information system is one representation of work systems where the products are information in one form or another (Alter 2001), e.g. a reimbursement system of an insurance company or tracking system for package delivery (Alter 2002). Often, an information system is a part of non-IS work systems, where its purpose is to support one of more of such systems (Alter 2002).

WSLC model pertains that systems are continuously changing and that the changes in them follow certain stages in a certain order. The phases of WSLD include operation and maintenance, initiation, development and implementation (Alter 2008a) (see FIGURE 1).

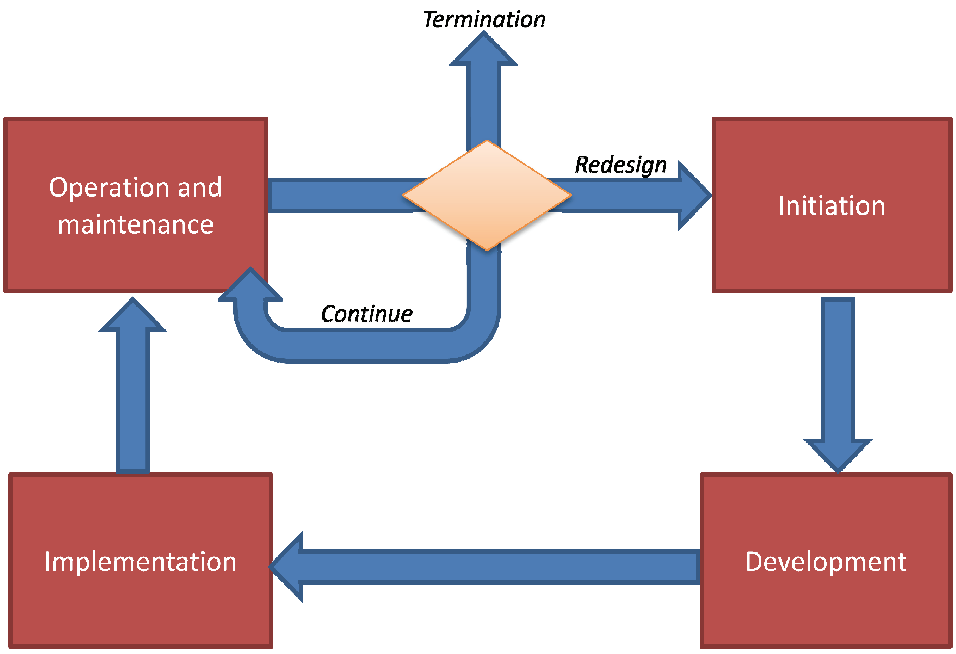

FIGURE 1 IS life cycle (adapted from Alter 2008a)

WSLD is a framework within which the IS evolves. According to the WSLC model the IS life cycle consists of periods of punctuated change, i.e. initiation, development and implementation of a new or renewed system, and stable periods during which the fundamental structures of the work system are not changed, i.e. operation and maintenance phase. A new punctuation is initiated when there is a need for a major change of the work system (Alter 2008a). WSLC model differs from ISD models and other linear life cycle models in that it recognizes activities after the development project and incorporates the aspect of continuous evolution. IT also views the IS life from organizational perspective. Organization is defined as an organization that utilizes the IS in its operations. The variety of IS life cycle definitions and the attractiveness of WSLC led to the formulation of the first tentative research question:

"What is the life cycle of an IS from organizational point of view?"

Well managed evolution is a precondition for enterprise system (ES) success (Markus and Tanis 2000). The importance of software evolution as the means for continuous and progressive change (Lehman and Ramil 2003) has been widely recognized in software engineering (see e.g. Adolph 1996, Boehm 1988, Bennett 1994, Bennett and Rajlich 2000). IT artefacts evolve by means of maintenance and renewal. IT artefact renewal is an episodically occurring (Lyytinen and Newman 2008), planned effort (Alter 2008a) to shift the system on a higher service level. Renewal is a mechanism through which an IT artefact changes in order to comply with the changing requirements of an organization. The renewal alternatives include system upgrading and modernization, the former typically dealing with packaged systems (see Seacord, Plakosh and Lewis 2003) and the latter with legacy systems (see Bennett 1994). Replacement occurs when the existing system has decayed beyond saving with the aforementioned methods and the old system is replaced with a new system (Seacord et al. 2003).

IT artefact renewal differs from the acquisition of a new system in the following terms: 1) the organization is familiar with the system (Olson and Zhao 2007), 2) the choices of the past define path dependency which influences the choices of the future (Lyytinen and Newman 2008), 3) the costs and benefits are lower than of new IT investments (Mukherji, Rajagopalan and Tanniru 2006) and 4) the customer has a long-term business-relation with the system vendor (Verville, Bernadas and Halingten 2005, Markus and Tanis 2000). Renewal is necessary in order to keep the IT artefact in compliance with the organization's needs and take advantage of technological opportunities (Mukherji et al. 2006), maintain IT aligned with business (Antikainen and Pekkola 2009) and to avoid misfits between the IT and organization (Strong and Volkoff 2010).

The decision making that starts a new cycle is particularly interesting because it provides a point of continuation between evolutionary cycles. The beginning of a new cycle is preceded by decision making: to continue the system use and maintenance as-is, to terminate the system or to initiate a major revision (Alter 2001). The last alternative starts a cycle. During the decision making process, the feasibility of the different evolution options are assessed. This includes benefit evaluation and assessment of the best possible timing.

IT artefact renewal investment means putting resources into an effort that aims at renewing the current IT artefact and maintaining or raisin the quality of IT services by upgrading or replacing outdated systems (Ross and Beath 2002). The owner of the renewal investments is the chief information officer (CIO) (Varghese 2003). The strategic objective of IT renewal investments varies between short-term profitability and long-term growth while their technology scope is mainly focused on shared infrastructure (Ross and Beath 2002). IT artefact renewals are significant investments for organizations. A business case description enables the prioritization between evolution alternatives and provides indication how the expected benefits will be realized (Peppard, Ward and Daniel 2007). According to Ward, Daniel and Peppard (2008) an IT business case consists of:

- *Business drivers* – what triggers the need for change in form of the proposed investment?
- *Investment objectives* – what are the intended achievements, the goal, of the investment?
- *Benefits* – what are the expected benefits of the proposed investments after all the objectives are met? Who are the benefit owners? On what

level the benefits occur (individual, group, organizational)? How are the expected benefits measured?
- *Project costs* – what are the costs of the proposed investment (including purchases, development, infrastructure, business change and on-going costs)?
- *Risks* – what are the risks (technical, financial, business and organizational) that could prevent the realization of the expected benefits?

According to Ward et al. (2008) business drivers define the investment objectives, which in turn originate the expected benefits. This led to the second tentative research question:

"What are the determinants for IT artefact renewal decision?"

In order to investigate the conditions leading to IT artefact renewal decision, three topics are subjected under further study: IT renewal triggers, timing and benefits. A trigger is defined here as a condition or factor that indicates a *need for change* of the IT artefact. A benefit signifies the *utility* received from a change of the IT artefact. Timing refers to the selection of an *appropriate moment* to conduct the desired change. An analysis of these determinants provides information for the decision making on IT artefact renewal. Of these three, timing is not explicitly included in a traditional business case model, but its inclusion in the study scope justified because it has direct implications on the benefits and it is related to the triggers (see Mukherji et al. 2006). There are other determinants that also influence the IT artefact renewal decision, such as risks and costs (Ward et al. 2008), but they are not included in the scope of study here.

The objective of this dissertation is to study the holistic IS life cycle and the determinants that lead to the renewal of IT artefact, the core of IS (Benbasat and Zmud 2003) from organization's perspective. The term organization refers to an organization that *uses* information systems in order to support its functions. Organizations that produce or vend IT applications or software are not included in the definition of organization here. Life cycle theory (Van de Ven and Poole 1995) and works system life cycle (WSLC) (Alter 2008a) are used as research frameworks. Life cycle theory describes how things change through predefined stages. WSLC model approaches IS life as a continuum of events that repeats in organizational context. It provides a framework that binds together the triggers, benefit evaluation and timing with the IS life cycle.

The thesis is based on two assumptions. Firstly, it is assumed that the understanding of the IS life cycle and renewal mechanics provide a useful concept for IT management and IT artefact renewal decision making. Secondly, the thesis is based on the assumption that the WSLC model represents the IS life cycle more realistically than the traditional ISD life cycle or other linear life cycle models when viewed from the organizational perspective. The thesis proposes that the beginning of a new cycle is determined by triggers, benefits and timing.

The rest of the thesis is organized as follows. Theoretical background is provided in chapter 2. The research methodology and research questions are described in chapter 3. Summary of the included papers is given in chapter 4. Chapter 5 presents the answers to the research questions and the contributions. In chapter 6, the conclusions, limitations, reflections, and directions for future research are given.

2 THEORETICAL BACKGROUND

In this section, the central concepts are defined and the theoretical background of the thesis is presented.

2.1 Information system and IT artefact

Various definitions for information systems exist (see e.g. Alter 2008a, Avison and Fizgerald 2008, Davis and Olsen 1985). In general, information systems (IS) consist of information, technology, and users that together aim at performing tasks in an organization. Organizations define the context for IS research (Wu and Saunders 2003); and it is precisely the context where the societal implications of IT are reflected upon. An information system can be viewed from functional or structural perspective (Hirschheim, Klein and Lyytinen 1995, p. 11):

> "From a structural perspective (see e.g. Davis and Olsen 1985), an information system consists of a collection of people, processes, data, models, technology and partly formalized language, forming a cohesive structure which serves some organizational purpose or function. From a functional perspective […], an information system is a technologically implemented medium for the purpose of recording, storing and disseminating linguistic expressions as well as for the supporting of inference making".

Here, the structural perspective is adopted and the information systems are defined as socio-technical systems (Lyytinen and Newman 2008) including social elements and technical elements. In Leavitt's (1965) diamond model, IS consists of task, structure, people, and technology (Leavitt 1965) (FIGURE 2). Task describes the purpose of the organization, i.e. producing products or services to its customers. Structure refers to the communication, authority and work flow practices. People refer to actors working in the organization to fulfill the organization's task. Technology is a means for problem solving by machines or programs.

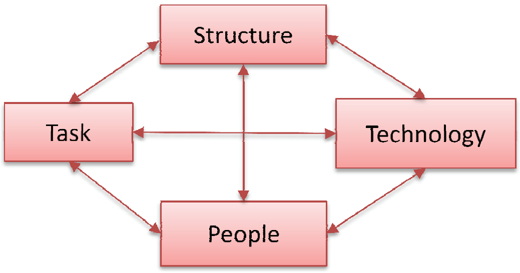

FIGURE 2 Leavitt's (1965, p. 1145) socio-technical model

These elements are dependent on each other and their relationship is symbiotic, like Alter (2008a, p. 449) describes it

> "Remove the system that is being served and the IS becomes meaningless. Remove the information processing and the larger system grinds to halt".

In this thesis, IS is defined as a work system (Alter 2008a) (see FIGURE 3). Work system treats IS as a socio-technical construct, as suggested by Wu and Saunders (2003), where social system and technical system are inseparable:

> "A work system is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific products and/or services for specific internal or external customers. An IS is a work system whose processes and activities are devoted to processing information, that is, capturing, transmitting, storing, retrieving, manipulating, and displaying information. Thus, an IS is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce informational products and/or services for internal or external customers." (Alter 2008a, p. 451)

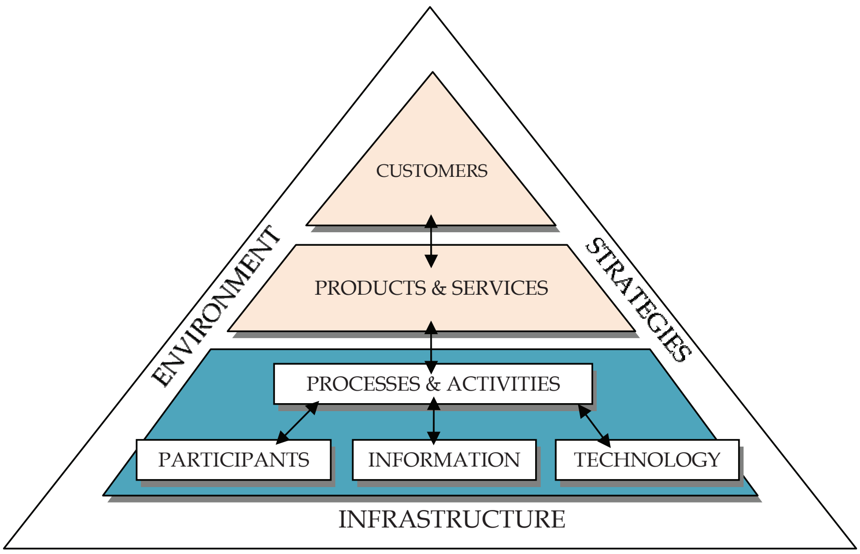

FIGURE 3 The work system framework (Alter 2008a, p. 461)

IS, as a work system, is formed by four elements, i.e. technology, information, participants, and processes and activities, form the core of a work system (Alter 2008a). The human *participants* are people who work in the business processes (Alter 2002). They may or may not be IT users (Alter 2008a). *Information* is created and used by the participants in the work system. It can be captured in computer systems or it can be verbal or tacit knowledge that has not been recorded or stored (Alter 2002). *Technologies* include various technologies (not only information technology) for general of specific business purpose (Alter 2008a). *Processes and activities* include e.g. workflow, decision making, communication, coordination, control and information processing. They connect the participants, technology and information together.

The following elements are not part of the information system but are included in the framework because they are directly related to it either by producing or using its results or providing the organizational context in which the IS is situated (Alter 2008a). *Products and services* are a combination of goods that the work system produces for its customers. They include for instance physical goods, information, services, social products (e.g. organizational arrangements) or intangible products and services (Alter 2008a). *Customers* are "the direct beneficiaries of whatever a work system produces, plus other customers whose interest and involvement is less direct" (Alter 2008a, p. 466). They can be internal or external customers. *Infrastructure* refers to the all of the resources that an operational works system requires, e.g. human resource, information resources or technology resources (Alter 2002). Infrastructure can be shared with other work systems and managed from the outside of the work system. The works system operates in an *environment* formed by organizational, cultural, competitive, technical and regulatory aspects. *Strategies* guide the choices and decisions within the work system. They include e.g. work system strategy, business strategy and enterprise architectures (Alter 2008a).

This thesis focuses on the technology component of an information system (Turban and Volonino 2010). The definition of Benbasat and Zmud (2003, p. 186) for IT artefact as the technology component of IS is adopted:

> "the application of IT to enable or support some task(s) embedded with a structure(s) that itself is embedded within a context(s)" [that] "encapsulates the structures, routines, norms, and values implicit in the rich *contexts* within which the artefact is embedded".

The IT artefact does not include in itself the structures and routines yet it reflects their norms and values (Benbasat and Zmud 2003). The position of IT artefact and IS in the works system framework is depicted FIGURE 4.

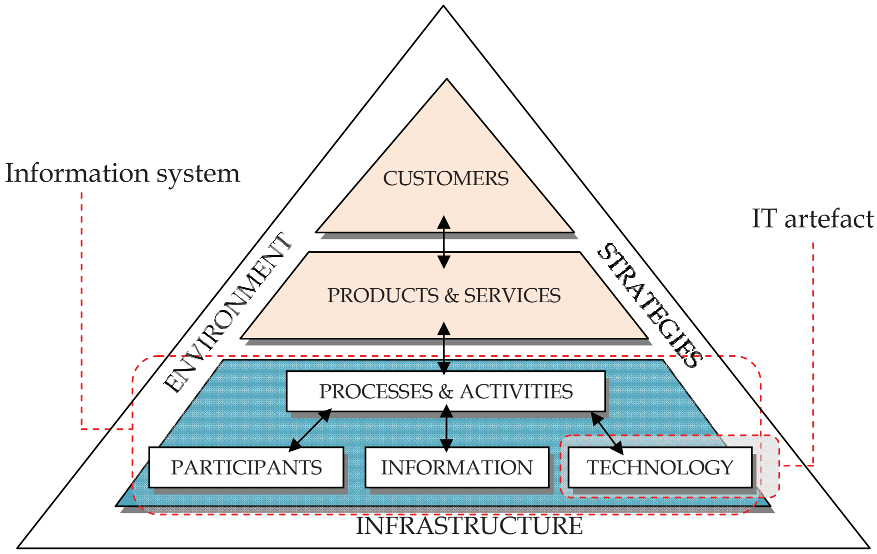

FIGURE 4 Information system and IT artefact in the work system framework

Any IT application that enables or supports some organizational function or task can be considered as an IT artefact (see Iivari 2007). These include, e.g. ERP systems, enterprise systems (ES), accounting systems and payroll system. Also operating systems and groupware or communication systems can be classified as IT artefacts.

2.2 Information system change

The *life cycle theory* explains the organizational processes of change and development as organic growth through specific stages (Van de Ven and Poole 1995). These stages are passed through in a predetermined order thus forming a sequence (Campbell and Reese 2002). Because this sequence of stages is in a form of a cycle, by definition, the same stages repeat over and over as reaching the end of the sequence (or a cycle) a new one starts (see Devaney 2007). The actual stages of life vary between different organisms yet the basic principle is the same. For instance, human life cycle consist of infant, child, juvenile, adolescent, and adult (reproductive stage) stages (Bogin and Smith 1998) while the one of bird's has stages of egg, incubation, hatching, chick-feeding period, growth, fledging period, independence, breeding, laying the first egg (Bennett and Owens 2002). One cycle consists of stages from the birth of an organism to the moment when the organism gives birth to a new organism, i.e. a generation.

A central element in the life cycle of any given entity is *change*. The change adopts different forms. A human organism changes, for instance, by growing: an adult man grows from a baby with significant changes in its appearance and functions. Likewise, a bird comes into the world from an egg, and after the fluffy-feathered infantine period develops into an adult avian. According to Lyytinen and Newman (2008, p. 590), the change in information systems

> "covers the generation, implementation, and adoption of new elements in an organization's social and technical subsystems that store, transfer, manipulate, process, and utilize information."

That is to say that the change can occur in either in the technical or in the social component of the IS, or in both of them. This change re-defines the IS elements and their relationships, i.e. it re-configurates a work system (Lyytinen and Newman 2008). A change in one element cascades changes in the others (Leavitt 1965) as articulated by Keen (1981, p. 25):

> "Leavitt's classification of organizations as a diamond, […] in which **Task, Technology, People**, and **Structure** are interrelated and mutually adjusting, indicates the complex nature of social systems. […] When **Technology** is changed, the other components often adjust to damp out the impact of the innovation".

Besides the aforementioned socio-technical view, IS change incorporates multiple other perspectives through which the change can be defined. Davis and Olsen (1985) propose that systems accommodate strain by changing their *form,* which takes place by *structural* changes, or *process*. They argue that structural change is triggered, for instance, when more terminals are installed in distant locations and there is an increased demand for data shareability. As a result the physical structure (hardware) of the IS changes. A process change occurs, for instance, when a computer system is changed to respond to the need for better data processing efficiency (Davis and Olsen 1985).

In Lewin's (1952) model, change occurs through *unfreezing, change* and *re-freezing* stages. Unfreezing signifies organizational preparation for change, which consists of disturbing the organizational equilibrium which is a prerequisite for the adaptation of a new order. This is followed by the change that moves the organization towards a new desired state of equilibrium. When the desired state has been achieved, is it re-frozen. Lewin's model thus suggests that a shift from one equilibrium stage to another necessitates "unlocking" the present state, changing it, and then locking it again on a new level.

The *punctuated equilibrium theory* explains the radical change dominating all life in history (Gould and Eldredge 1977). Punctuated equilibrium theory is founded on the assumption that instead of gradual evolution nature exhibits rapid changes of short duration. These radical changes are called punctuations. Between the changes is stasis; a relatively long period of time (compared to the duration of the punctuation) of tranquility with little changes (Gersick 1991). The rate at which the punctuations occur is called relative frequency (Gould and Eldredge 1977). The periods of stability and revolution are explained through deep structures, i.e. abiding underlying order (Gersick 1991). The deep structure is the object of change during the revolutionary periods yet, simultaneously, it constrains the change (Gersick 1991).

> "*Deep structure* is the set of fundamental "choices" a system has made of (1) the basic parts into which its units will be organized and (2) the basic activity patterns that will maintain its existence" (Gersick 1991, p. 14).

Deep structures have different parts which form units which again perform "work" by exchanging resources with the environment through which they organize and maintain themselves (Gersick 1991). This description surprisingly resembles the core of a work system. Strong and Volkoff (2010) relate enterprise system deep structures to functionality and data. They contrast deep structures to *surface structures* that are related to system interfaces and usability. Radical change can occur also in surface structures. For instance migration to service oriented architecture (SOA) adds a new surface layer while leaving the deep structures intact (see e.g. Vitharana, Bhaskaran, Jain, Wang and Zhao 2007).

Lyytinen and Newman (2008) presented a *punctuated socio-technical information system change model* that proposes that there are two types of change within IS life cycle, namely incremental change and radical change, that occur in a socio-technical setting. The incremental change refers to the need of a system to continuously adapt to its environment during the periods of tranquillity (Lyytinen and Newman 2008). In their theory, Lyytinen and Newman (2008, p. 595) emphasize the role of change triggers, which they call gaps or critical incidents:

> "Occasionally, when any one component becomes incompatible with others due to increased variation (e.g. malfunctioning, learning, replacement) we can observe a structural misalignment, which we label here a gap – a property of a system that affects the systems' behavior and its repertoire of responses. A gap is any contingency in the system which, if left unattended, will reduce the system's performance and threaten its viability. Often events that generate gaps are abrupt: a system failing, a financial crisis, or key people leaving."

In work system a change that creates a gap can take place at any of the work system levels (Alter 2008a):

- *Processes and activities*: change of the degree of structure of processes and activities, level of integration, complexity or rhythm.
- *Participants*: new skills, change of incentives or change of social relations.
- *Information*: use of different information of providing information in a different form than before, or at different level of detail.
- *Technology*: re-configuration, upgrading or replacement of technology.
-

A gap can occur between any of the IS elements: task, technology, structure, people (Lyytinen and Newman 2008). In this thesis, the focus is on the technology component change. The possible technology component gaps are: technology-structure gap (GAP 1), technology people gap (GAP 2) and technology-task gap (GAP 3) (FIGURE 5).

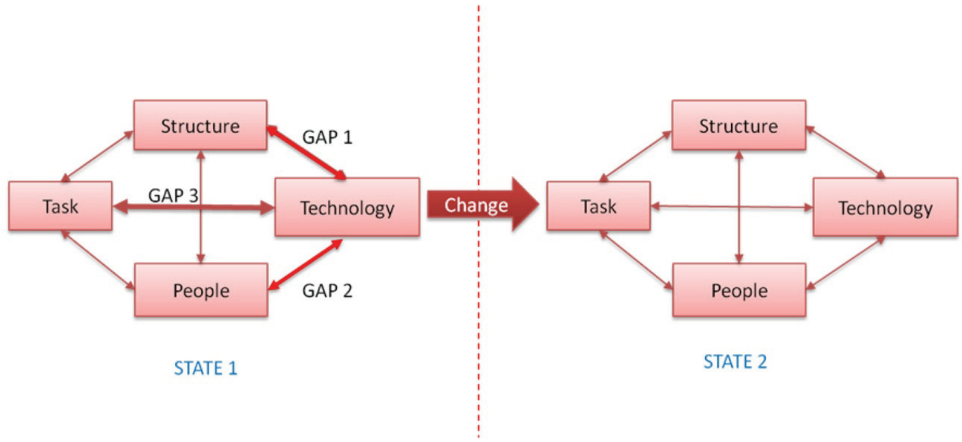

FIGURE 5 IS change (adapted from Lyytinen and Newman 2008).

IS change is a result of a change in any of the IS components alone or their combinations (Leavitt 1965). A system compensates a gap in two ways: by incrementally adapting on a component level according to the rule of the deep structure, or by changing the deep structure (Lyytinen and Newman 2008). A gap can also trigger a cascade of subsequent punctuations (Lyytinen and Newman 2008). Successful gap-compensation yields a new state where the equilibrium between the IS components is restored (Ahmad, Lyytinen and Newman 2011).

2.3 IT artefact evolution

In this section, the terminology related to IS change is defined concerning the concepts of evolution, renewal, maintenance and replacement.

Evolution is result of natural selection where favourable variations are preserved and continue their existence and injurious variations are rejected by nature itself (Darwin 1968, p. 131). In IS context, the natural selection means that systems that are able to comply with the changing requirements of their environment survive, and other will die out. Evolution occurs globally and locally. Globally it can be seen e.g. in ERP systems; how they have changed from inventory control packages in the 1960´s, through material requirements planning (MRP) systems in the 1970s, manufacturing resource planning (MRP II) systems in the 1980s to the enterprise resource planning (ERP) systems in the 1990s, and finally, to the extended ERP of the 2000s (Rashid, Hossain and Patrick 2002). The evolution of World Wide Web (WWW) (Conolly 2000) or the transition from custom-made IS to commercial software-based systems in organizations (Sawyer 2001) or the evolution of user interfaces from character-based to graphic (see e.g. Myers 1998) are also manifestations of global scale evolution. At organizational level, IT artefact evolution is limited to an IT population within an organization or a group of organizations sharing the same IT applications.

Software engineering defines evolution as *continuous* and *progressive change* (Lehman and Ramil 2003) covering a wide range of activities ranging from evolutionary development (Adolph 1996, Boehm 1988) to large scale migration projects of legacy systems (Bennett 1994, Bennett and Rajlich 2000). From user-organization's point of view, information systems evolve as new capabilities and functionalities are added and the system's cover area is increased through the extended applications (Esteves and Pastor 1999). The evolution of commercial enterprise information systems can include the implementation of modules that have been purchased with the initial ERP installation but never taken into use (Beatty and Williams 2006).

Information systems as a socio-technical systems change through two primary means: by incremental, gradual adaptation (the first order of change) and by revolutionary, episodic change (the second order of change) (Lyytinen and Newman 2008). The change can occur in planned or unplanned form or through their combinations (Alter 2008a, 2008b). The degree of change varies from daily maintenance to large scale development projects depending on magnitude of the harmful state of the system, the rate at which its performance will deteriorate and to what extent it poses a threat for the system survival (Lyytinen and Newman 2008). Planned changes are conducted via formal, episodic projects, such as renewal or replacement projects. Unplanned changes occur spontaneously and without formal projects in form of small adaptations and experimentations. Maintenance corresponds to the first order change by its nature. It can be either planned or unplanned (Alter 2008a). The relation between the above mentioned evolution taxonomy and the change types is presented in TABLE 1. No literature reference was found to describe unplanned second order change.

TABLE 1 The change orders of evolution

Change order	**Planned**	**Unplanned**
1st order change	Maintenance	Maintenance
2nd order change	Renewal Replacement	-

In this thesis, IT artefact evolution is defined as a result of any planned or unplanned change through which the IT artefact adapts to its environment. The change activities that result evolution of an IT artefact consist of maintenance, renewal and replacement (FIGURE 6).

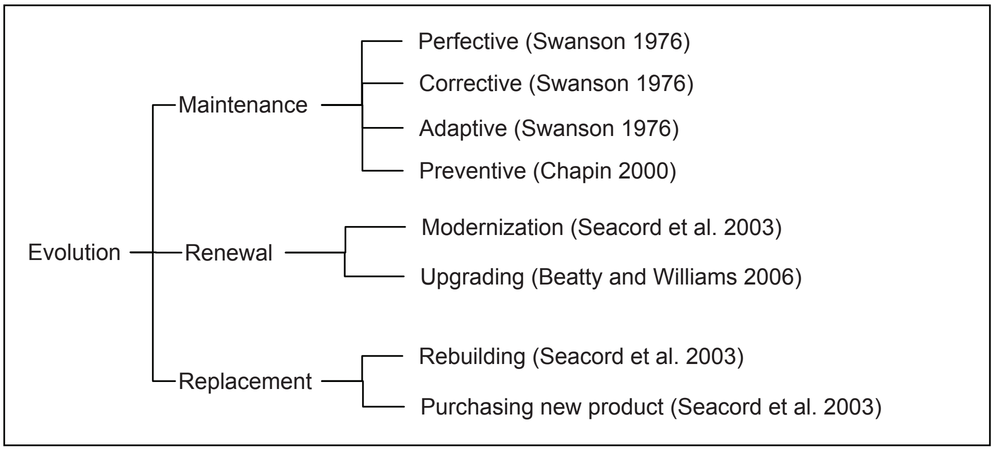

FIGURE 6 The taxonomy of IT artefact evolution

IT artefact evolution is a sequence of maintenance, renewal and replacement activities (FIGURE 7). The inclining dotted line represents the growing business requirements for the system's functionality. The solid black line indicates the system functionality that increases with small steps during periods of maintenance and with big step representing modernization activity. Together these two lines indicate how well the system corresponds to the business needs.

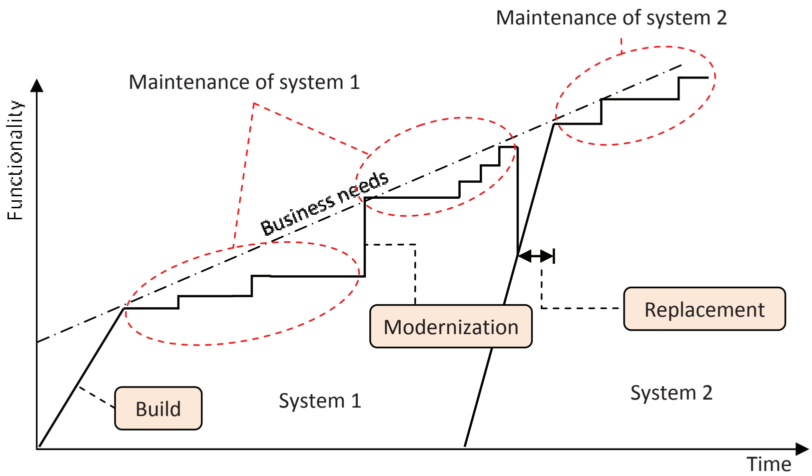

FIGURE 7 Software evolution (Seacord et al. 2003, p. 8)

Maintenance is the means of keeping the IT application operational and in compliance with its environment with small modifications and changes made to the system. The four maintenance categories are:

1) *Perfective maintenance*. The system is enhanced by improving its usability, performance or compliance with user requirements (Swanson 1976).
2) *Corrective maintenance*. Defects or errors are repaired (Swanson 1976).
3) *Adaptive maintenance*. The system is changed to comply with the changes occurring in its technology environment (Swanson 1976).
4) *Preventive maintenance*. Future maintainability and reliability are anticipated by changes that seek to prevent problems of future maintenance proactively (Chapin 2000, Seacord et al. 2003).

The maintenance of packaged software takes place by means of releases or service patches. A release is a software package that is delivered between the major versions and in order to make small scale additions to functionality (Kremers and van Dissel 2000). Service patch contain corrections of errors (Kremers and van Dissel 2000).

Renewal means restoring something to a new or fresh condition (MOT Collins English Dictionary 2006). Renewal refers to large scale, structural changes, adding new capabilities to (Weiderman, Bergey and Smith 1997) or expansion of the current system (Olson 2009). The changes are made at the architectural level and they increase the economic and strategic value of the software (Weiderman et al. 1997). Renewal includes, for instance, implementing a distributed architecture or a graphical user interface (Seacord et al. 2003). Renewal takes place through modernization (Seacord et al. 2003) or upgrading (Beatty and Williams 2006). Modernization includes various techniques including retargeting, revamping, source code translation, code reduction, and functional transformation, program modularization and data re-engineering (Seacord et al. 2003) that are used for renewing legacy systems (Adolph 1996). Compared to maintenance, modernization means larger scale changes while still conserving a large portion of the system (Seacord et al. (2003). Upgrading means the change of the running version of packaged software. A version means an improved software package that has significantly more functionality than its successor and possible uses new technology (Kremers and van Dissel 2000). Here, Microsoft's definition for upgrade is used as suggested by Khoo and Robey (2007, p. 556):

> "An upgrade is a software package that replaces an installed version of a product with a newer version of the same product".

Replacement means that the existing system, whether custom-made or commercially purchased, is terminated and replaced with another system (Seacord et al. 2003). The replacing system can be either built from the scratch, i.e., rebuilding the old system, or purchasing a new commercial product. For instance enterprise and ERP systems have been increasingly used to replace legacy or proprietary systems (Davenport 1998, Esteves and Pastor 1999). However, an opposite trend has also emerged: organizations have aborted their ERP system implementation and returned to the old custom-made IS (Esteves and Pastor 1999).

2.4 Life cycle models

Life cycle models describe change in IS through developmental stages. Often this is depicted with linear sequence of stages. The basic ISD process (see FIGURE 8) includes feasibility study, system investigation, system analysis, system design, implementation (construction), and review and maintenance (Avison and Fitzgerald 2008).

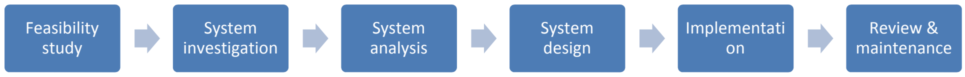

FIGURE 8 ISD life cycle

In case of commercial systems, the linear life cycle models emphasize system acquisition over the development. These include for instance Esteves and Pastor's (1999) ERP life cycle framework (see FIGURE 9), ES experience cycle (Markus and Tanis 2000) and ERP system life cycle management model (Chang et al. 2008). These models attempt to describe the whole life cycle yet they fail in putting the emphasis on the initial development or acquisition project, while leaving the rest of the IS life under "evolution" (see Esteves and Pastor 1999) or "expansion" stage (see Chang et al. 2008). Verville et al. (2007) have adopted an iterative approach for the representation of the ERP life cycle with emphasis on the acquisition. This model suggests that the life cycle phases of an ERP system are acquisition, maintenance and implementation (Verville et al. 2007). They emphasize the acquisition phase due to the related costs, benefits and risk assessment and challenges for the organization. Yet, the task of purchasing new ERP system is, however, only one task in the continuum of life stages of an IS.

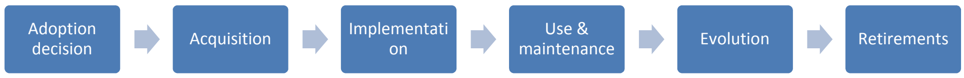

FIGURE 9 The ERP life cycle framework

A more holistic approach can be found in the work system life cycle (WSLC) model (FIGURE 10) (Alter 2001). The work system is a general representation of a system that produces goods or services. Information system is one representation of work systems where the products are information in one form or another (Alter 2001), e.g. a reimbursement system of an insurance company or tracking system for package delivery (Alter 2002). Typically, an IS is a part of non-IS work systems, where its purpose is to support one of more of such systems (Alter 2002).

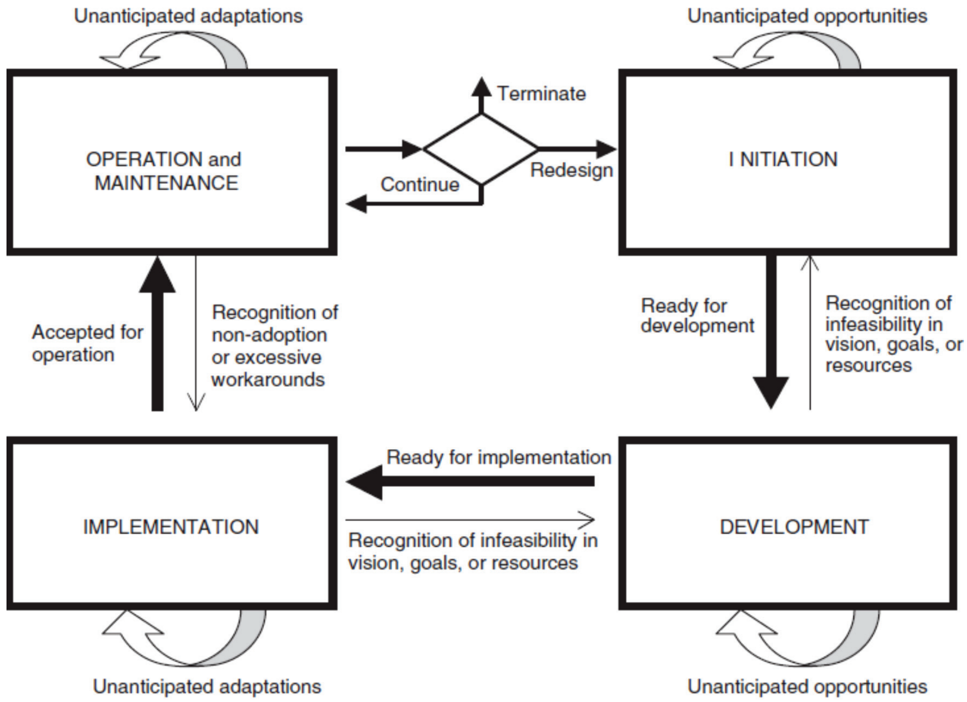

FIGURE 10 The Work System Life Cycle (Alter 2008a, p. 467)

WSLC model has been revised many times (Alter 2001, Alter 2002, Alter 2003, Alter 2008a). *Operation and maintenance* includes performance monitoring, evaluation and correction of small errors, improvement of work practices and keeping the system both organizationally and technically operational (Alter 2003). The system is supported and maintained operational (Alter 2001). *Initiation* phase includes the visualization the renewed IS, definition of operational goals, resource allocation, definition of time frames for the renewal project and economic assessment of the intended renewal project (Alter 2001). The objective of the assessment is to elicit a technologically, organizationally and economically feasible solution for the perceived problem (Alter 2003). *Development* phase includes requirements specification, development, modification or acquisition and configuration, hardware and software installation, documentation, preparation of training materials, testing. As a result, the renewed IS is technically operational and ready for the implementation of the planned change in the organization (Alter 2003). *Implementation* itself includes implementation method selection, planning, change management, training, system conversion from old to new or renewed, and acceptance testing. As a result, the new or renewed system is organizationally operational (Alter 2003). The result of one round of the life cycle is called a major revision, iteration or generation (Alter 2001).

A work system evolves through iterations of both planned and unplanned change (Alter 2002). Planned change manifests in the form of a designated project with necessary planning and resource allocation. Unplanned change, on the contrary, means gradual, small changes and adaptations within the work system that occur naturally without the establishment of a designated project organization. The *unanticipated opportunities,* related to the initiation and development phases, include the "uncontrolled" projects that, contrary to traditional ISD view, are positive rather than negative phenomena with respect to the overall compliance between the IS and the organization. The thick arrow between the phases (e.g. Recognition of infeasibility in vision, goals, or resources) signifies that each phase is an input for its successor phase (Alter 2001). The returning thin arrow implies the possibility to return to the previous phase if need be (Alter 2001). Alter (2008a) indicates that the WSLC model represents an iterative punctuated change, occurring when a renewed (or new) work system is implemented. This is followed by a stable period during which the fundamental structures of the work system are not changed. A new punctuation is initiated when there is a need for a major change of the work system.

The WSLC model can be criticized for closely resembling the traditional ISD process, i.e. it focuses only on the development of the technical element of the IS. The terminology is derived directly from the ISD context, e.g. "development" for the actual measures taken for the IS renewal.

Also, there are unexplained areas in the model that leave space for free interpretation. The *unanticipated adaptations* remain without further clarifications or practical examples. The diamond shaped figure between operation and maintenance and initiation phases lacks explanation. Intuitively it refers to a renewal *decision making*: whether to continue the operation and maintenance of the system, to terminate it or to initiate a new cycle in system's life by renewing it. In an earlier version of the work system model (Alter 2001, p. 4) this decision is explained as part of the operation and maintenance phase as follows:

> "To express the fact that many systems go through major revisions, the model's operation and maintenance phase includes a recurring decision about whether to continue maintenance and incremental improvements (continuous change), whether to undertake a major revision by starting a new iteration of the four phases (discontinuous change), or whether to terminate the system."

Despite of the critique that Alter's work system model received during the IT artefact debate (see Wu and Saunders 2003) the WSLC is eminently suited for modelling the life cycle of an IT artefact. Similarly, the life cycle theory has been criticized of being insufficient for designing complex information infrastructures (Hanseth and Lyytinen 2004). However, the life cycle theory is particularly suitable for *understanding* them. Particularly interesting is the decision point between the evolutionary cycles because it is a precondition for the start of a new, planned cycle in IS's life.

2.5 IT artefact renewal triggers

Disequilibrium between IS elements leads to change in order to re-establish equilibrium (Lyytinen and Newman 2008). Technology related disequilibrium occurs between technology and other IS elements. A gap between technology and task means that the technology is inadequate or wrong for the task given. A technology-people gap occurs when the actors are unable to adapt, operate or use the technology. A technology-structure gap takes place when technology is misaligned with a given structure. The technology related gaps act as triggers for IT artefact change. (Lyytinen and Newman 2008)
IT artefact upgrade decisions are a result of interaction between certain *motivating forces* and *contingency forces* (FIGURE 11) (Khoo and Robey 2007). Motivating forces push or pull the upgrade decision forward, i.e. they give a reason motivation for the software packages renewal (see Beatty and Williams 2006). There are two sources for motivating factors, namely, internal requirements, which spring from the organization itself, and the external requirements such as dependence on vendor (Khoo and Robey 2007). Internal requirements crystallize in business needs, IT needs and risk mitigation; external requirements culminate on dependence on vendor and include software functionality and technical support. The contingency forces represent forces that inhibit, oppose or delay the upgrade decision. The most significant contingency force is the availability of internal resources (Khoo and Robey 2007) i.e. capital, IT expertise, management focus and capacity for change (Ross and Beath 2002). The upgrade decision is a result of a combination of motivating and contingency forces (Khoo and Robey 2007). The available resources seldom trigger upgrade on their own, but without them, the upgrade cannot be done despite of the urgency of the motivating factors.

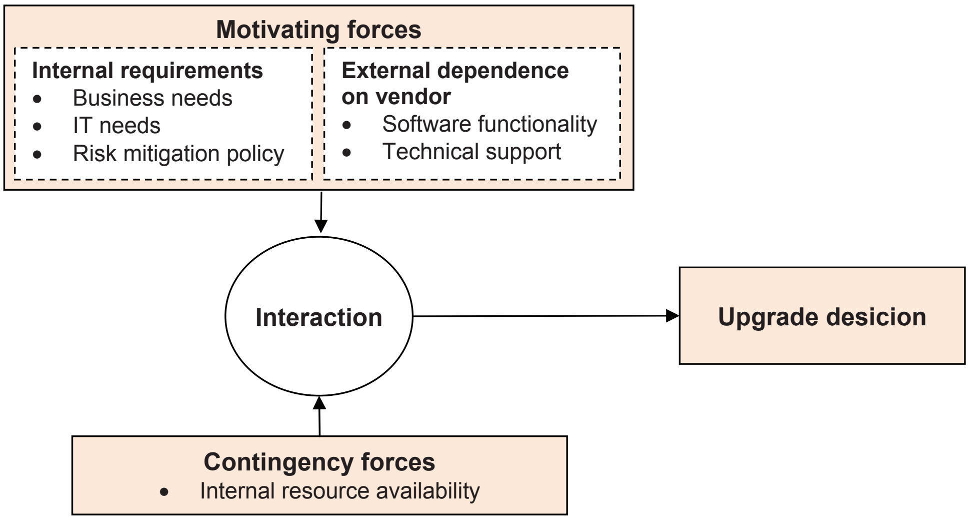

FIGURE 11 Upgrade decision model (Khoo and Robey 2007, p. 563).

ERP system upgrade is typically triggered by the need for new functionality, end of vendor's support for the running version (Olson and Zhao 2007, Kremers and van Dissel 2000, Khoo and Robey (2007) (see TABLE 2). An organization that has purchased and uses packaged software is in a tight relationship with the software vendor. In this relationship the customer organization is dependent on the vendor in regard of software changes, additions, corrections and software support (Beatty and Williams 2006). Due this dependency, the system vendor has a significant influence on software upgrade decisions of its customer's (Beatty and Williams 2006) and the initiation of packaged software upgrade is commonly due to vendor's influence (Khoo and Robey 2007).

TABLE 2 ERP system upgrade triggers

ERP system upgrade triggers	Author
• desire for new functionality • the looming end of vendor's service • desire for better vendor support • fixing errors • module integrations	Olson and Zhao (2007)
• added (new) functionality • compliance with standards • expired or impending end of support • keeping the system up-to-date • dissatisfaction with the technical performance of the current system	Kremers and van Dissel (2000)
• business growth • need for more flexibility • need for more functionality • avoiding unstable software • continuous vendor support • availability of resources	Khoo and Robey (2007)

The average time for development of a new software version has significantly shortened within the last two decades. In the early 1990s, the average time for the emergence of a new version was three years while in the 2006 it varied between 18 and 24 months (Beatty and Williams 2006). This is directly reflected at the pressure the vendors put on their clients concerning software upgrading. ERP software providers actively persuade and pressure the organizations to upgrade their application suites to newer versions by setting a "de-support day" (Beatty and Williams 2006) or "sunset date" (Khoo and Robey 2007) after which they are not supported by the vendor any longer. Customers that want to have the support service are consequently pushed to upgrade before the sunset date in order to get the desired service (Beatty and Williams 2006). Customers that take care of the maintenance themselves or otherwise choose to rely on unsupported software are running a risk of having an incompatible system in the future (Khoo and Robey 2007).

In order to avoid difficulties with system upgrades Damsgaard and Karlsbjerg (2010) give two exemplary scenarios to be avoided. In *Blind alley* scenario an organization is using packaged software that is losing its market share to competing commercial packages. As a consequence, the user organizations have to abandon the system and acquire another replacing system. *One-way street* is a scenario where an organization is facing a challenge with very narrow set of alternatives for system changes. There are only few alternatives for upgrading or adding functionality but the organization is locked in with the vendor due to the high switching costs for another product by another vendor. These scenarios describe the *inhibiting* factors that limit the possibilities of business driven upgrade decision. Nevertheless they are strong determinants of upgrade.

2.6 The timing of IT artefact renewal

Beatty and Williams (2006) suggest that about six to 12 months after the initial ERP system implementation arises the first pressure wave to add new modules in it. This is closely related to the first signs of benefit realization after the initial investment. After that, the organization will face the question of upgrading on regular basis making the timing of IS renewal important. IT upgrades are resource intensive, time consuming and laborious (Kremers and van Dissel 2000). An organization hence carefully considers how often and when the IS is renewed in order to avoid abundant stress and load on the organizational members (Kremers and van Dissel 2000). However, once the customer adopts an ERP system the upgrading becomes an unavoidable and inevitable part of their joint future (Khoo and Robey 2007). As Kremers and van Dissel (2000, p. 55) point out

> "The question is not "*Should* we migrate?" but "When *do* we migrate?"

Timing is about finding a balance between "too soon" and "too late". Upgrading too early involves risks such as adopting immature and faulty software (Mukherji et al. 2006). Postponing the upgrade for too long, on the other hand, increases the difference between the version in use and the vendor's latest supported version which leads to increasing difficulties in the upgrade (Mukherji et al. 2006). A commonly used strategy for IT upgrade timing is *leapfrogging*: the most economical way of upgrading is to wait until there is a need for change, skip unsuitable software versions, and acquire a clearly beneficial version when such becomes available (Mukherji et al. 2006). Timing is defined as a strategic decision concerning the optimum time to conduct IT artefact renewal. Scheduling is operational level activity which aims at allocating resources, including time, within a project.

The timing of IT artefact renewal is dependent on available resources (Kremers and van Dissel 2000, Khoo and Robey 2007). Lack of financial resources leads to postponing renewal. Other ongoing or planned projects may reserve organizational resources from upgrade projects leading to delayed upgrading (Khoo and Robey 2007). Also, if the payback period of the earlier renewal (or upgrade) project is not finished new renewal projects may be halted (Kremers and van Dissel 2000). Company policy on upgrading frequency may dictate the timing of renewal (Kremers and van Dissel 2000). Vendor influences the upgrade timing by positive and negative means. Positive means includes the offering of new versions and actively promoting upgrading. Negative reinforcement of timing comes from the sunset date that defines the last possible upgrade time (Khoo and Robey 2007). The timing of IT renewal is also influenced by the business's high and low seasons. Almost every product is seasonal (Radas and Shugan 1998). This defines the business strategy and annual business calendar with its *seasonal patterns* (Radas and Shugan 1998). It is obvious that during a busy season resources are allocated for making business, and unnecessary other projects, such as IT projects, are avoided. Other seasonal variations, including the closing of the accounts in the end of the fiscal year and annual reporting intuitively also influence the timing of the IT projects in general.

The timing of IT projects is typically managed with *portfolio analysis* (Seacord et al. 2003, p. 29-30), which maps on-going and future IT projects. By analysing this portfolio of IT investments the management can assess the urgency of each investment proposal and their priorities with respect to each other (GAO 1996). Other methods include real options analysis (see e.g. Campbell 2002), information economics (Parker, Benson and Trainor 1988) or capital investment appraisal techniques (Milis and Mercken 2004).

2.7 IT artefact renewal benefit assessment

IT investments differ from "normal" business investments due to their high uncertainty (Olson 2009), risks, and the management's deficient knowledge about the IT investments since they are not core business but have a supporting role (Remenyi, Money and Sherwood-Smith 2000, pp. 35-37). Uncertainty deals with issues of what we know and what we do not know; i.e. decisions with consequences that cannot be predicted (Tannert, Elvers and Jandrig 2007). Risk represents the probability of an undesired event to take place (Holton 2004). In practice, IT upgrades are costly (Ng 2001, Olson and Zhao 2007, Kremers and van Dissel 2000), difficult and time consuming (Kremers and van Dissel 2000). However, the rules for the IT investment justification are the same as for any other business investment (Remenyi et al. 2000, p. 34).

The justification of IT investments in general is a sum of various factors including strategic, tactical, operational considerations, and tangible and intangible benefits (Turban and Volonino 2010). Tangible benefits are associated with quantifiable and economic benefits. They are savings or earnings that directly influence the company profitability and are visible in the accounts

(Remenyi et al. 2000, p. 7). Intangible benefits are the soft or qualitative improvements that have great value but are difficult to detect in the company accounts or profit (Remenyi et al. 2000, p. 7). These include e.g. user satisfaction, IS quality and effectiveness (Remenyi et al. 2000, pp. 153-158). Ideally the evaluation of the IT investment takes place at two instances: before and after the project. The purpose of the *ex ante* evaluation is to assess or *predict* the anticipated benefits prior to the initiation (Remenyi et al. 2000, p. 25). The role of the *ex post* evaluation is to verify, *prescribe*, the realization of the expected benefits (Remenyi et al. 2000, p. 25).

In a well-constructed business case the expected benefits are derived from investments objectives which, in turn, are derived from investment drivers (Ward et al. 2008). If the main driver for renewal investment is, for instance, vendor's end of support for the current version (Olson and Zhao 2007, Khoo and Robey 2007, Beatty and Williams 2006, Kremers and van Dissel 2000, Ross and Beath 2002), the investment objectives could be:

- upgrading the system to a version that is supported by the vendor
- ensuring the continuity of system support
- minimizing risks of running a non-supported version.

And, the related benefits could hence be:

- responsibility of the system support remains at vendor's
- safe system operation and functionality by the continuing support
- savings on IT staff resources and risk avoidance.

Other possible desired benefits from ERP system upgrade include solutions for design weaknesses or programming errors, and new or improved features (Olson and Zhao 2007).

The IT investments are costly and the top management naturally expects something back for the invested money and resources. Traditionally, the benefits have been expected to manifest in financial terms as *return on investment* (ROI) (Dekleva 2005). Other traditional investment appraisal methods including payback, cost-benefit analysis, net present value (NPV), internal rate of return (IRR) (Irani and Love 2002). In fact, the amount of investment assessment methods available for IT investment evaluation is extremely large (see e.g. Andresen 2002).

In practice, the allocation of benefits for the investment objectives is a challenging task. Relying on financial benefit assessment only is risky because it allows "creative" calculations, unrealistic assumptions, understating the costs of organizational implementation (Ward et al. 2008). ROI, as a benefit assessment approach for IT investments, has been heavily criticized in the IT investment research (see e.g. Dekleva 2005). It is unable to capture the value of IT investments and cannot ensure in any way that a proposed IT investment is in compliance with the business needs (Varghese 2003). Empirical findings give similar indications, as demonstrate the following words of an ERP project leader:

> "We didn't convert the benefits into money, because everyone can calculate them as he wants, I can show you that our ROI is 200% or 300% if I want" (Haddara and Päivärinta 2011, p. 7).

One study reports that the IT investment decisions are largely based on inadequate data and undefined decision criteria on costs, risks and return (GAO 1996). The same study reveals that the IT investments funding was decided upon the judgmental expertise of the decision makers (GAO 1996). The use of traditional economic evaluation methods such as payback, cost-benefit analysis, NPV, IRR (Irani and Love 2002) has other shortcomings too. They only capture the quantifiable aspects of an investment, while excluding the qualitative and value-based aspects that cannot be converted into financial figures (Ward et al. 2008). They seek to predict the impacts of IT investments at organizational level, e.g. by measuring the organizational impact of the IT investment such as sales growth, return on assets, return on net worth, the ratio of information-processing expense, total operating expense, increased productivity, and the investment costs (see e.g. Mahmood and Mann 1993), while the actual benefits may not be traceable with those measures. Traditional project evaluation tools are also insufficient in dealing with the high uncertainty which is more than typical for IT investments (Schwartz and Zozoya-Gorostiza 2000).

Companies are facing major difficulties both in building a business case for IT projects (Ward et al. 2008). Haddara and Päivärinta (2011) recognized that organizations are unable to utilize the existing evaluation methods in their IT investment evaluation practices despite of the plentiful supply of available methods. This problem can be easily interpreted as a sign for a need for better evaluation methods. Empirical findings show however, that in reality, companies need is to improve their evaluation practices instead (Ward et al. 2008).

Overstatement of benefits is a problem related to the financial benefit assessment for gaining acceptance for the renewal investment. The predicted benefits presented in the business case are too optimistic and impossible to achieve (Ward et al. 2008). Farbey, Land and Targett (1992) found that majority of IT investments in organizations are justified by *ad hoc* way even if the organization had formal justification procedures to be utilized. These ad hoc decisions included act of faith, obedience to the orders of corporate headquarters, letting technology changes guide the decisions step-by-step, or must do justification for survival (Farbey et al. 1992).

Companies are not doing the ex post evaluation of the realization of the anticipated benefits (GAO 1996, Dekleva 2005). However, the evaluation of the IT investment decisions is relevant because the decision of today influence the decisions of tomorrow, and the decision making and evaluation practices are inherited in the organization in good and bad (Rose, Rose and Norman 2004).

Ex post benefit evaluation is not carried out in companies as commonly believed (Haddara and Päivärinta 2011). The expected benefits from ERP systems are self-evident, the evaluation methods are costly and difficult to use, and the rationality of the methods is not trusted by companies (Haddara and Päivärinta 2011). Additionally, government's support for ERP investments seems to debase the motivation for evaluating benefit realization (Haddara and Päivärinta 2011).

Nevertheless, the management of resource allocation for ERP projects still remains as an issue. The self-evident nature of the ERP project outcomes is traced back to the attitude towards IT in general: the ERP system is seen as a commodity. This study concerns new ERP system investments and not explicitly renewal or upgrade projects. However, it provides a baseline for the investigation of benefit realization practices in IS renewal projects.

In order to spur the more inclusive and rigorous benefit assessment, Ward et al. (2007) suggest using a mixture of benefits in their naturally different forms. This can be done in association with the business case development framework that supports the differentiation and structuring of benefits. The *degree of explicitness* of the benefits refers to the amount of knowledge that is available prior to the investment. *Financial* benefits include all the benefits that can be expressed in financial terms. In the end of the evaluation these will provide the basis for calculating the overall value of the investment proposal. *Quantifiable* benefits are benefits that can be measured and have an established measurement device in the organization (or it can be easily installed) with which the magnitude of the benefits can be estimated. This is prediction of the future benefits in quantifiable terms. *Measurable* benefits are aspects that have an identified measure and are currently measured (or can be easily measured). This allows them to operate as a baseline for further investment benefit evaluation. *Observable* benefits are benefits that can be valued by individual perception and opinion. They are also called subjective, intangible or qualitative benefits. The mechanism in the framework is that one positions firstly all the benefits in the observable row. As the amount of reliable evidence (measurable, quantifiable or financial) emerges the benefit moves upwards in the matrix. The changes addressed by each expected benefit are classified in three groups: The investment yields benefits that allow the organization, its groups, or individual 1) do new things that were not possible before, 2) do things better than before, or 3) stop doing things that are no longer necessary. The assignment of benefit owner is critical. Benefit owner harvests the benefits of the investment and hence provides the value for the benefit. It also induces the benefit owners to engage in the IT project and add more weight to the business case. (Ward et al. 2008) The ownership of IT renewal project, as a whole, is with the CIO, whose responsibility is to guard the service quality of the shared IT infrastructure (Varghese 2003). The formal justification of must-do investments is not relevant as such, yet the comparison of different alternatives in must-do situation is relevant (Remenyi et al. 2000).

3 RESEARCH METHODOLOGY

This section outlines the research objective and questions. The research process, framework and methodology are described. The relation between the included articles and research process and research framework are explained. Data collection and analysis methods are described. Finally, the validity of the research is evaluated.

3.1 The research objective and questions

The objective of this dissertation is to study the holistic IS life cycle of and the determinants that lead to the renewal of IT artefact, the core of IS (Benbasat and Zmud 2003) from user-organization's perspective. This is done by investigating and analyzing the existing IS life cycle models and comparing them to the empirical findings of this thesis. The objective is based on the assumption that the life cycle of an IT artefact exhibits recurring events as it evolves. Based on this, the first research question (RQ1) is re-formulated:

> *RQ1: What is the life cycle of an IS from organizational point of view?*

IS change is complex because it is a sum of changes in the combination of IS elements, i.e. technology, information, participants and organizational processes and activities (Alter 2008a). This thesis focuses on the technology component (Turban and Volonino 2010) which is here defined as equal with the concept of IT artefact (Benbasat and Zmud 2003). Hence, RQ1 is clarified with a sub-question:

> *RQ1-1: "What is the life cycle of an IT artefact from organizational point of view?"*

Derived from life cycle theory (Van de Ven and Poole 1995), change is growth through specific stages that form a cycle. Hence, there is a point where one cycle ends and another begins. In IS context this point of continuation is interesting because each evolutionary cycle has economic consequences in the organization. Since IT artefact renewal occurs episodically in organizations (Lyytinen and Newman 2008) the conditions that lead to a decision to start of a new evolutionary cycle must also occur episodically. From this, the second research question (RQ2) was formulated:

RQ2: What are the determinants for IT artefact renewal in organizations?

In order to investigate the IT artefact renewal determinants more closely, it is necessary to break the determinants down on a more specific level. Hence, the second research question is divided into sub-research questions:

RQ2-1: What are the triggers for IT artefact renewal?
RQ2-2: What factors define the IT artefact renewal timing?
RQ2-3: How are the IT artefact renewal benefits evaluated?

The sub-questions enable a more detailed and coherent analysis of the different types of determinants influencing the IT artefact renewal decision in organizations.

3.2 The research scope

The triggers for change can originate from any of the IS elements and cause renewal respectively in any of the four elements. In this thesis the focus is on the renewal of the technical element. The work system model states that 1) an IS consists of four elements (processes and activities, participants, information, and technology), and 2) the WSLC depicts the life cycle of the whole IS (see Alter 2008a). Based on this, an assumption is made that all the elements follow the same life cycle separately (see FIGURE 12). This enables breaking down the IS elements in their own life cycles in order to direct the research focus on the IT artefact and related renewal determinants.

In this thesis the life cycle and evolution of the technology element of IS, i.e. the IT artefact, are studied. The remaining elements of IS, i.e. processes and activities, participants or information (see Alter 2008a) are not included in the study. The IT artefact renewal takes place within an organization. Hence, the study is limited to the perspective of the organization that uses the IT artefact. IT vendor perspective is not engaged. The focus is explicitly on the examination of the *conditions* leading to the decision making concerning IT artefact renewal rather than the decision process itself. The decision making occurs between operation and maintenance, and initiation phases of WSLC (Alter 2008a). The remaining phases, i.e. development and implementation are not subjected to study here. The focus is on planned IT artefact renewal. The unanticipated adaptations and opportunities recognized by Alter (2003) are not studied.

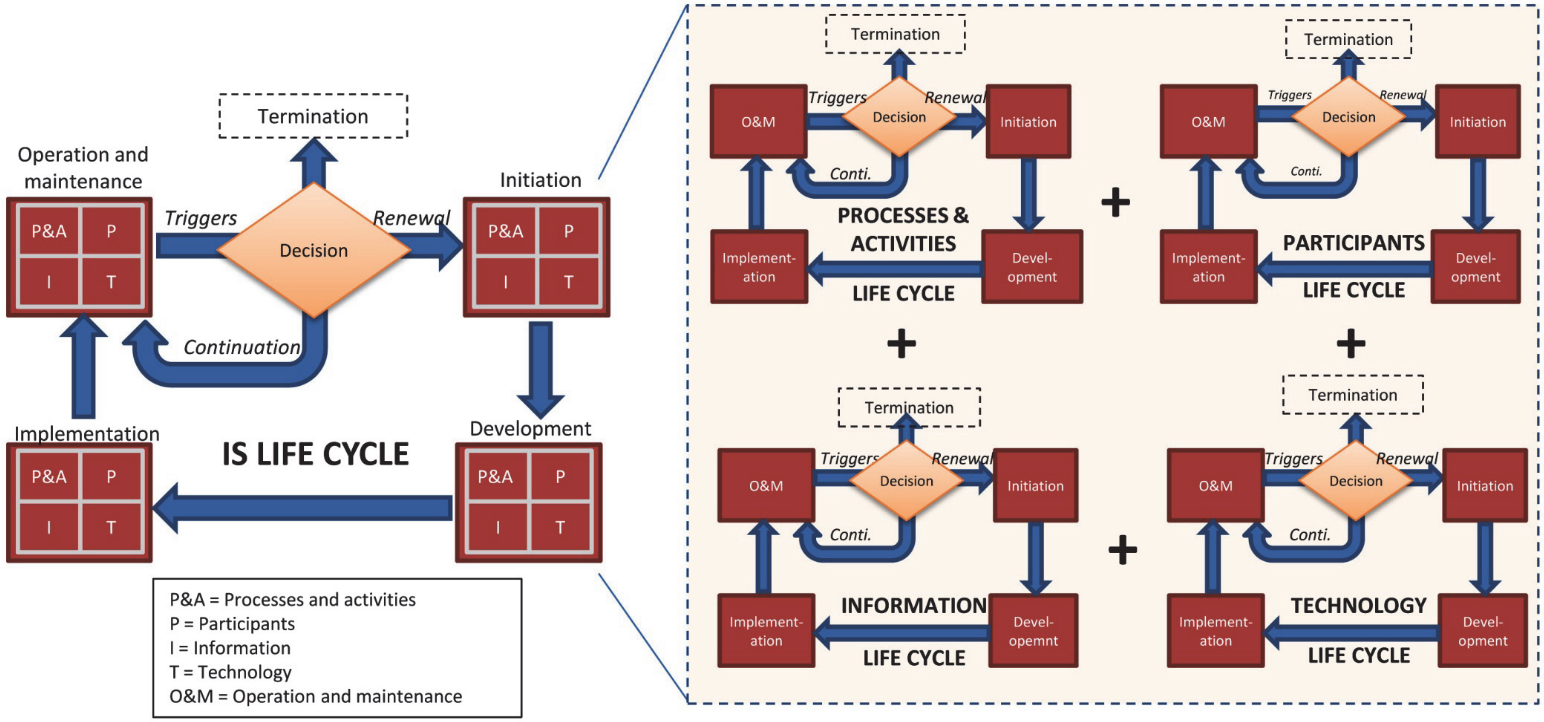

FIGURE 12 IS elements and their life cycles

The examination of the IT artefact renewal timing is limited to the study of the determinants of timing. An answer to a question *when* to renew is not sought. The benefit evaluation is covered for the purpose of understanding the IT investment evaluation challenges in general and bringing the evaluation to IT artefact renewal context. Benefit calculations are not conducted because the objective is not to study in-depth individual benefit evaluation methods but instead to reach a general understanding of how the renewal alternatives are evaluated. The research is not intending to answer to questions such as the productivity paradox (Brynjolfsson 1993) or any other questions seeking the relationship between IT investments and increased productivity on company or industry level. Custom-made systems and commercially packaged software are used as reference systems to study the IT artefact life cycle and its evolution through their maintenance, renewal and replacement. It is recognized that outsourcing strategies, such as acquiring the organizational systems as software as service (SaaS) (Greschler and Mangan 2002), through application service providers (ASP) (Turban and Volonino 2010, p. 597), or as open source software (OSS) (Olson 2009) are alternatives for system maintenance and upgrading (Olson 2009) yet their examination is excluded from the scope of this dissertation.

3.3 The research process

The research process generally unfolded from particular to general. It started from the practical need to address the challenge of IT renewal decision making, more precisely comparing alternative renewal options in order to produce reliable data on them to the decision makers in companies. Research gaps were identified in literature as suggested by Rudestam and Newton (1992). These originated research questions and motivated empirical research. Empirical evidence on them was then gathered from companies in order to answer to the questions, fill the gaps and, finally, to complement the existing theory base with refined concepts. The empirical findings, in turn, opened new avenues and indicated the direction for the next cycle of theoretical investigation. The research process ended with the theory formulation for IS life cycle and connecting it back to the original starting point – the evaluation of IT renewal benefits. The research process unfolded in three cycles in timely linear order (see FIGURE 13). The research process and final research questions evolved over time as the research proceeded. This gradual development of the research process mirrors the maturation and growth of the author as a researcher. It represents a learning process as well as research process.

The first research cycle consisted of reviewing the IT benefit evaluation literature and traditional economic investment evaluation methods in ELTIS research project. The goal was to study how is the return of IT renewal investment evaluated in practice as this was a relevant problem in the companies participating in ELTIS project. At this point the research was software modernization oriented. Literature review formed a theory base that suggested revision of the research problem: the challenge was not in the actual application of benefit assessment methods but instead the use of financial methods, including ROI, by default, and the identification of suitable methods from the seemingly endless ocean of benefit evaluation methods (see e.g. Andresen 2002, Farbey et al. 1992).

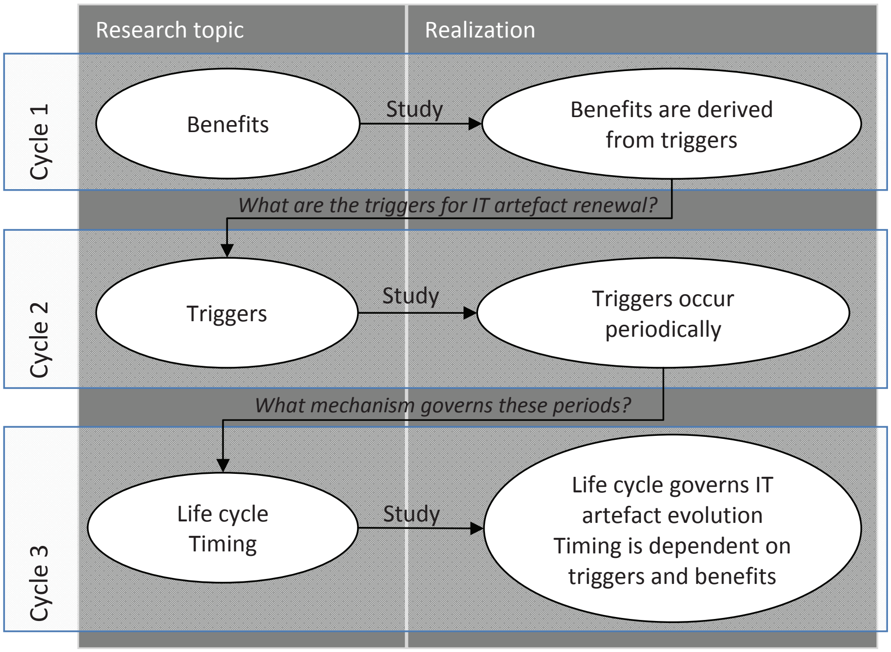

FIGURE 13 The research process

The research question of the first research cycle was RQ2-3 *"How are the IT artefact renewal benefits evaluated?"* As a result, a literature review and analysis was conducted in order to identify the challenges related to the use of financial benefit assessment methods in IT renewal context. Then, ISEBA framework was constructed in order to tackle the challenge of selecting a suitable benefit assessment method. ISEBA was tested with two participating companies during their evaluation process on legacy system modernization options and related benefits. A set of suitable methods were identified and applied accordingly. In case 2, two members of the research group, including the author, conducted the method selection together with the company representatives and also were responsible for carrying out the actual measurements and data analysis. Case 2 included both ex ante and ex post evaluation with the same evaluation criteria. The author did not participate in case 1 which was carried out simultaneously by other project members. Both cases were, however, discussed in project group meetings regularly and all project members were informed about them. The first research cycle resulted papers 5 and 6. From the cases arose the realization that 1) financial benefits are not the only determinant of software renewal and that 2) the benefit assessment method selection is dependent on the project goals, which in turn are dependent on the overall modernization (renewal) motives (triggers). This realization launched in motion the second research cycle.

The second research cycle started with the examination of the conditions leading IT renewal in literature. The research question was RQ2-1 *"What are the triggers for IT artefact renewal?"* Literature concerning this topic was not abundant and this motivated the design and formulation of an interview study on modernization and replacement cases (multiple-case study). The objective was to compare the differences and similarities between the factors that trigger modernization and replacement projects in companies. Other issues related to the differences and similarities of SW modernization and replacement cases were also included in the study but they were not included in this dissertation. This was the last study within ELTIS research project that ended 2006. The result was paper 3. The findings lead to a realization that renewal triggers are periodically occurring. The need for change accumulates during time as the system ages and at some point breaks the threshold of change need. In time, the renewed system also ages and subsequently triggers a need for renewal. The findings gave inspiration for further research: What is the mechanics that governs the IT renewal initiation beyond the case-specific triggers? Is it possible to identify larger a scale pattern that could explain the phenomena behind renewal triggers on a general level? How could this recurrence be modelled?

The third research cycle started with the investigation of IS life cycle in order to set a large scale framework for the IT artefact renewal phenomenon. The original software focus shifted towards IS due to its more comprehensive nature. Literature review focused, on one hand, on the renewal commercial software packages (e.g. ERP systems) and, on the other hand, on IS life cycle. Alter's (2008a) WSLC model was used as a framework for IS life cycle and for positioning the triggers, benefit assessment and timing in it. The literature review led to the realization of the importance of upgrade timing and lack of empirical research about it. It was also noticed, that IT artefact renewal is intimately related to a larger picture – the IS life cycle – and that the IT renewal and related decision making are recurring events in IS life cycle. The research questions RQ1 *"What is the life cycle of an IS from organizational point of view?"*, RQ1-1 *"What is the life cycle of an IT artefact from organizational point of view?"* and RQ2-2 *"What factors define the IT artefact renewal timing?"* were formulated. This was followed by philosophical, conceptual and analytical study of literature and related theories (e.g. life cycle theory and change theories) leading to investigation of whether the better understanding of the IS life cycle would yield better understanding of and solutions for the IT artefact renewal evaluation problem. This required empirical investigation. New data collection was thus designed and conducted in order to study IS life cycle and renewal timing in companies. The empirical findings provided evidence for the holistic IS life cycle as a mechanics that governs the IS evolution. The third research cycle resulted papers 1, 2 and 4, which hold the understanding of the IT artefact renewal as part of a bigger picture, and that the benefit evaluation, timing and triggers are all derived from that same big picture, and are influenced by the past choices and will influence the future choices.

3.4 The research approach and methods

The selection of the research method is based on the research problem. Because the IT artefact renewal is approached from organizational decision making perspective the problem area encompasses a set of different research areas, namely IS life cycle, IT artefact renewal triggers, timing and benefit assessment. As a consequence, a combination of research methods is applied respectively to study each of these areas. The use of more than one approach ensures triangulation, i.e. examination of the problem area with variety of tools and multiple sources in order to gain a reliable and truthful picture of it (Jick 1979, Yin 2003, p. 99). The results of these individual studies are material for the interpretative analysis and conceptualization that synthesizes the final results of the thesis.

In order to draw the final results of the thesis, interpretative research approach (Walsham 1995, Walsham 2006, Traut 2001) is used. Interpretative tradition is based on the notion of subjective idealism where *"Each person constructs his or her own reality"* (Walsham 1995). Interpretative approach aims at capturing the subjective perceptions of the informants and making interpretations of them (Walsham 1995). The research areas are investigated in different settings which can be considered as different cases in themselves. Hence, the overall research method of this thesis is interpretative case study (Walsham 1995). Case study, as a research strategy, is particularly suitable in studying *decisions* and *organizations* (Yin 2003, p. 12). Three separate cases can be identified, each approaching the research topic *IT artefact renewal* form different standpoint. Single setting (Lee and Baskerville 2003) approach is used. The research methodology is depicted in FIGURE 14.

The first setting, representing the first case, is the CIOs in Finnish corporations that utilize IS to support their operations. IS life cycle and IT artefact renewal timing is investigated by interviewing CIOs of large corporations. The research follows the principles of interpretative case study. The results of these investigations are reported in papers 2 and 4. The interview is based on theoretical-conceptual analysis, the results of which are reported in paper 1.

The second research setting consists of the key person participating in IT artefact renewal projects within Finnish corporations' IT management. In this setting, the research focus is on the factors that trigger IT artefact renewal in organizations. IT management representatives are interviewed by phone. Comparative case study method is used as IT artefact renewal (modernization) triggers are compared to IT artefact renewal triggers in organizations. The results of this study are reported in paper 3.

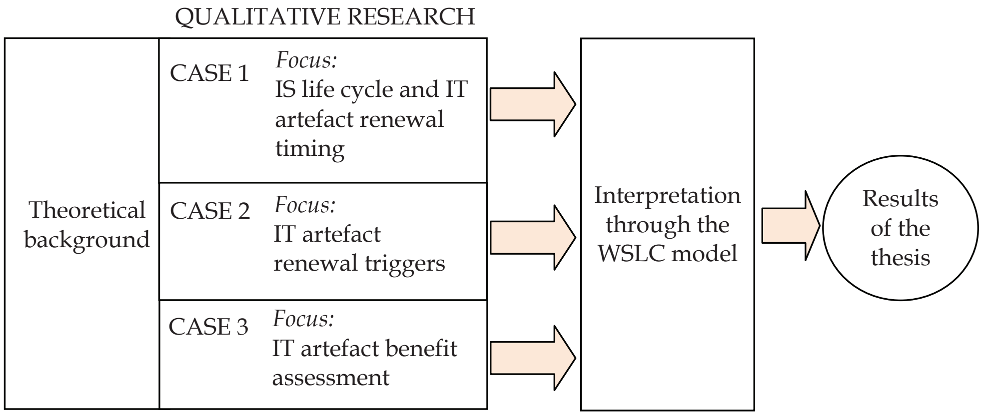

FIGURE 14 The research methodology

The third research setting, representing the third research case, focuses on IT artefact renewal decision making and related benefit assessment in organizational cases. Experiences from benefit evaluation are obtained through the use of a ISEBA framework that is constructed to support the benefit evaluation process in organizations. The results concerning the benefit evaluation are reported in paper 6. The building of the construct and conduct of the empirical cases is preceded by literature review, the results of which are reported in paper 5.

Theory is used as an initial guide to design the data collection (see Walsham 1995) and, later, as a guide for interpreting the results. Literature review is used in order to establish a baseline of the previous and current literature. It is used as basis to form the research questions. In the interpretation, the WSLC (Alter 2008a) is used as theoretical model. Conceptual analysis was used in the investigation of IS life cycle. Due to the nature of the research, critical approach (Traut 2001, Orlikowski and Baroudi 1991) the *status quo* of the way the life cycle metaphor is applied in IS field is needed. The correspondence between characteristics of living organisms and IS require solid theoretical base to justify the further steps of the research and the applicability of life cycle theory in IS context. The theoretical analysis also provides with research gaps and thus is a source of motivation for the research questions.

Empirical research approach is needed in order to connect the theoretical findings and concepts with real-life. Qualitative research methods are used for studying the issues that stem from the reviewed literature and theories in a deeper level in order to gain understanding and evidence on how they manifest in practice (e.g. IS life cycle or IT renewal benefit assessment). Qualitative research approach (Denzin and Lincoln 2000) is used in order to investigate the IS life cycle from user organization's perspective (RQ1) and the determinants of

IT artefact renewal (RQ2) empirically. Qualitative research is chosen because it enables the inquiry of in-depth views and philosophical argumentation on the studied phenomena (Denzin and Lincoln 2000). The target group (sample) is chosen appropriately according to the specifications of the research in question (Stake 2000). The interview, the primary data source of interpretative cases studies, is used as data collection method because it allows the best position for the researcher to obtain the perceptions the subjects have on the research topic (see Walsham 1995). The research topic has been sparingly studied previously and thus collecting of descriptive data from various organizations by means of interview was considered the best approach. The interview questions were based on previous research, literature and theory published on refereed journals. The content coverage of the interviews is validated by peer-reviews, i.e. the research associates review the interview questions prior to the interviews, or they are rigorously assessed in the research group.

3.5 The validity of the research

The corroboration of evidence in qualitative research is done in order to ensure that the material has not been fabricated, discounted or misinterpreted (Dey 1993). As the main research method is interpretative case study, the validity of the research has to be in accordance with the quality measures of interpretative approach and of case study. The quality of case studies, as well as any other empirical social research, can be determined against the following validity measures: construct validity, internal validity, external validity and reliability (Yin 2003, p. 34). Validity "is the degree to which the finding is interpreted in a correct way" (Kirk and Miller 1986, p. 20). According to (Walsham 1995) Yin’s description of case study method includes both positivist and interpretative features. Consequently, Yin’s (2003) validity measures for case studies are adopted here from interpretative perspective leaving the inapplicable measures, such as internal validity, out.

Construct validity describes how well the selected research design succeeds in capturing and representing the studied phenomenon (Yin 2003, pp. 34-35). Construct validity can be increased by using multiple sources of evidence, maintaining the chain of evidence and having the subjects review the results (Yin 2003, p. 36). Construct validity of the research is ensured with the use of multiple sources of evidence (Sawyer 2001, Yin 2003, p. 36) or triangulation where the evidence on the research topic is inquired from many sources (Yin 2003, p. 99). Triangulation reduces the odds of misinterpretation as data is collected from many sources instead of one (Stake 2000). The advantage of the use of multiple cases is that it provides better grounds, better accuracy and higher degree of generalizability than a single case (Eisenhardt and Graebner 2007). This is obtained by using in three different research settings, each one belonging to a different research cycle. Each research cycle approached the studied phenomenon from slightly different perspective and with slightly different method. For instance in the second cycle IT managers were interviewed by phone on concerning their experiences on software modernization and replacement projects, while in the third research cycle CIOs were interviewed in person concerning the IS life cycle. The chain of evidence was maintained in the collection and analysis of the data in order to be able to return to the chain of logic that led to certain conclusions or possible generalizations as suggested by Yin (2003, p. 34). Construct validity was also ensured by delivering the results of the data analysis to the interviewees (see Yin 2003, p. 34). No contradicting comments or change requests was returned.

Internal validity deals with the legitimacy of chain of action-consequence and their explanations in explanatory or causal studies (Yin 2003, pp. 36-37). Since the purpose of this study is not searching causal relationships, internal validity was not concerned with in this research as suggested Yin (2003, p. 34). Also, even though Yin (2003) relates internal validity to case study validity measures it is essentially a validity measure of positivist research (see Straub, Boudreau and Gefen 2004).

External validity refers to the generalisability of the case study results (Yin 2003, p. 37). Orlikowski and Baroudi (1991) argue that interpretative approach does not seek generalization from the research setting to a population but instead understanding of the studied phenomenon and its deep structure. In other words, interpretative research seeks generalization within cases, not across cases (Lee and Baskerville 2003). Based on this notion, the results of this thesis are generalisable within the three research settings described in the previous section. The results provide the reader with indication of how the CIOs of large Finnish corporations perceive IS life cycle or IT artefact renewal timing, or which antecedents were considered as triggers for IT artefact renewal by IT management key personnel. The resulting increase of understanding "can then be used to inform other settings" as suggested by Orlikowski and Baroudi (1991, p. 5) but not generalized as rules that would apply in different settings or in a larger population. Also, the perceptions of CIOs' on IS life cycle cannot be generalized as a factual definition of IS life cycle because they are subjective perceptions bound to time and context and interpreted by the researcher.

According to Yin (2003, p. 38), The external validity of interpretative case studies does not lie in their generalization to other cases but instead how well the findings can be generalized to a theory. This "type ET" (from empirical statements to theoretical statements) generalization holds that findings from empirical research, whether experimental or case study, are inputs for a theory formulation (Lee and Baskerville 2003). In this thesis, the findings from each research setting operate as input for theoretical statements concerning the relationships between the findings themselves and the relationship between the findings and the WSLC model. However, as Lee and Baskerville (2003) suggest, the resulting theory cannot be generalized as such beyond the given cases. Hence, the results are not generalisable to a larger population of CIOs in a wider selection of countries, for instance. Such generalizations can be made only after the theory has been sufficiently tested in different settings (Yin 2003, p. 37).

Reliability gives indication whether the study has been fabricated or its results falsified (Dey 1993), or whether the results can be trusted. In this study, reliability is being assured by giving a description, within the limits of confidentiality, of the research design (including the research process, subjects and data collection methods), so that it is transparent and the readers could have an understanding how the research is done and compare it to other cases as suggested by Stake (2000). In case study research, reliability measures how well the same case research could be repeated and would the results be the same (Yin 2003, p. 37). However, the repetition of the exactly same study would be impossible in practice, since qualitative research always takes place in a specific time and space as suggested by Dey (1993). Likewise, the interpretations made by the researcher are subjective and most likely not repeatable (see Stake 2000). Even the replication of the same study would require an inconvenient amount of resources and efforts as often is the case in qualitative research (Dey 1993). Hence, the reliability of this thesis is largely based on the researchers' research ethics. Joint efforts in included papers also strengthen the reliability.

4 INCLUDED PAPERS

This section provides a brief summary of the papers included in the dissertation. Each of them stands an individual research published during the research process. The summary includes a description of research objective, research methods, content and main results. Papers 1, 3, 4, 5, and 6 are results of joint research efforts with other researchers. The contribution of the author in these co-authored papers is described in the following summaries.

4.1 Paper 1: The Life Cycle of Information Systems

Kankaanpää I and Pekkola S (2011) The life cycle of information systems. Manuscript, University of Jyväskylä. Submitted to Journal of Information Technology Theory and Application (JITTA).

The first paper presents a theoretical foundation for the concept of IS life cycle. The objective was to compare the characteristics of an IS to the general characteristics of life in order to find out whether it can be considered as a living entity and whether the use of life cycle metaphor can be justly applied in IS context. The study is based on conceptual examination of the life cycle metaphor from biology and its suitability in IS context. Due to the nature of the research objective, the conceptual-analytical research method (Järvinen 1999, p. 13) was chosen. The biology-based concept of life cycle was used as a framework to investigate in what extent the IS life follows the natural cycle of life. The existing life cycle models were examined and analyzed in the light of the biology-based life cycle definition. Philosophically this research shares both *ontological* and *teleological* aspects (Hirsjärvi, Remes and Sajavaara 2010, p. 130). The ontological objective was to explore the true nature of IS by contrasting it to the biological life characteristics and life cycle. The teleological questions concerned the purpose of this exploration: the analysis was carried out in order to justify the further application of life cycle metaphor in IS context. This research adopted critical approach (Traut 2001, Orlikowski and Baroudi 1991) because it questions the *status quo* of the way the life cycle metaphor is accepted and commonly used in IS field.

The conceptual analysis reveals that IS, as a socio-technical system, shares the commonly accepted characteristics of life. An IS has a hierarchical structure, which provides the basis for the controllability of its complexity; it has inbuilt homeostasis that keeps the operation within optimum (or tolerable) limits; it has inheritance on multiple levels (technical, organizational, group, individual) in form of knowledge transfer; a good correlation between form and function (business-IT alignment) is a prerequisite for the survival and an indicator for change need of an IS; the "reproduction" of IS is needed for future generations to emerge; an IS has to adapt to the changing organization and external demands and it evolves during time; it grows and matures through the life stages; it requires external energy (in form of human efforts and other resources) in order to stay alive, i.e. operational; and it has to response to external stimuli (changing requirements) in order to remain in use (i.e. alive). The IS life cycle was defined as a sequence of stages from implementation of an IS to the implementation of an upgraded IS of subsequent generation. In IS literature the best corresponding model is the work system life cycle that treats IS as a socio-technical, constantly evolving system (Alter 2008a). The results of the conceptual analysis indicate that the life cycle metaphor is suitable in IS context when IS is viewed as a socio-technical system. It depicts the iterative nature of IS life and provides a holistic conceptual framework for understanding the continuously changing nature of IS from user organization's perspective. Further analysis revealed, that the life cycle of both custom-made and commercial packaged software is similar. The differences are noticeable e.g. in the development goals, user participation, the number of stakeholders throughout the life cycle, evolution mechanisms and renewal triggers.

This paper was planned and written by Irja Kankaanpää and Samuli Pekkola. The literature survey, analysis and theory formulation was conducted by Irja Kankaanpää.

4.2 Paper 2: Information system's life cycle: Interpretative study on CIO perceptions

Kankaanpää, I. (2010) Information system's life cycle: Interpretative study on CIO perceptions. Submitted to Scandinavian Journal of Information Systems (SJIS) 6.7.2011.

The second paper approaches IS cycle from IT management point of view. Life cycle theory (Van de Ven and Poole 1995) was used as a framework to guide the research. Literature review covers different types of life cycle models on IS field including ISD process (Avison and Fitzgerald 2008), ERP lice cycle models (e.g. Esteves and Pastor 1999) and work system life cycle model (Alter 2008a). The objective was to investigate the perception of CIOs on IS life cycle. Empirical study was carried out. Interpretative research approach (Walsham 1995) was adopted. Fifteen CIOs from various fields of industry were interviewed in order to gain empirical evidence on the nature of IS life cycle in user organizations.

The findings indicate that the CIOs perceive the IS life cycle in three different ways, namely, as a linear sequence of stages, as system age or as a part of the life cycle or organizational processes. The findings also indicate that the perception on IS life cycle is related to the perception of IS, as suggested earlier by Orlikowski and Iacono (2001). The interpretative analysis revealed two underlying perception towards IS. When viewed from technical perspective, IS consists of a software and hardware. It represents to the CIOs a technical system with a clearly defined sequence of life stages. It lives in the organization and has a clearly defined life span from birth to death. Moreover, it can be replaced, changed, integrated or outsourced. When viewed from socio-technical perspective, however, the IS it is tightly embedded in the organizational processes and IT infrastructure, incorporates human-aspect and shares the purpose of the organization and its tasks. On the other hand, the CIOs perceive the IS life cycle as a continuously evolving socio-technical system that lives with the organization, processes, tasks and with the work that humans carry out in the organization. The study revealed that renewal of IS as a technical system is triggered by the need to change or add something new to the existing system. These are determined by business, technology, and vendor. Business defines the framework in which the IS operates and projects its needs, whether external or internal origin, on the IS. Technology both guides IS evolution by its tendency to grow old and eventually turn the system legacy if not renewed. System vendor commonly triggers system renewal by halting the provision of system support and recommending version upgrade.

This study was planned and conducted by Irja Kankaanpää.

4.3 Paper 3: Legacy system evolution – A comparative study of modernization and replacement initiation factors

Kankaanpää I., Tiihonen P., Ahonen J. J., Koskinen J., Tilus T., and Sivula H. (2007) Legacy System Evolution – A Comparative Study of Modernisation and Replacement Initiation Factors. In A. Costa, V. Duarte, B. Encarnação, L. Marques and V. Pedrosa (Eds.) the Proceedings of the Ninth International Conference on Enterprise Information Systems (ICEIS 2007), DISI (pp. 280-287), Madeira: INSTICC.

The third paper studies the factors leading to the initiation of software renewal by means of system modernization or replacement. The objective was to investigate how modernization and replacement projects differ from each other with respect to their initiation triggers. On this little studied field the possible results were difficult to hypothesize in advance and it required pioneer research to open up the research field. Hence qualitative research was applied for the empirical investigation of the phenomenon. 29 upper and middle level IT managers were interviewed. Each interview was focused on a modernization (14 cases) or a replacement case (15 cases).

The findings indicate that the most common initiatives for modernization are the business development, system's old age and obsolete technology. The most common initiatives for system replacement, on the contrary, are the old age of the existing system, the end of vendor's support and system's inability to respond to company's business needs. In general, however, system age, obsolete technology and high operation or maintenance costs are drivers for both modernization and replacement decisions. The findings also indicate that there is a correlation between the intended business development and modernization. Organizations that modernized their legacy system exhibited a clear desire for business process development. The interviewees were aware of IT trends in their field of business and decided to modernize their system(s) in order to enhance their business. The findings indicate strong vendor influence on decisions to replace a legacy system with a new system. A major motivator for system replacement was vendor's announcement of ending the system support.

This paper was planned by Irja Kankaanpää, Päivi Tiihonen, Jarmo J. Ahonen, Jussi Koskinen, Tero Tilus and Henna Sivula. The design of the data collection was carried out by all the authors. Data collection was conducted by Päivi Tiihonen, Henna Sivula and Irja Kankaanpää. Data analysis and writing of the paper was done by Päivi Tiihonen and Irja Kankaanpää.

4.4 Paper 4: Timing the information system upgrade

Kankaanpää I. and Pekkola S. (2010) Timing the information system upgrade. In P. M. Alexander, M. Turpin and J. P. van Deventer (Eds.) the proceedings of the 18th European Conference on Information Systems (ECIS2010) (pp. 1545-1557). Pretoria: University of Pretoria.

The fourth paper studies packaged system upgrade timing from user organization's perspective. The objective was to identify conditions that define the timing of upgrading. This phenomenon was approached by means of literature survey followed by empirical investigation. Literature survey covered customer's and vendor's interests with respect to software product upgrading. It revealed potential conflicts between customer and vendor with respect to upgrade timing. The customer's interest is to upgrade the IS on the need-basis while the vendor's interest is to induce upgrades based on their own product life cycle. Interview questions were formulated on the basis of these literature findings and interviews conducted accordingly. Fifteen CIOs were interviewed face-to-face in order to canvas their thoughts, reasoning and experiences related IT renewal timing. The empirical study was carried out in accordance to interpretative tradition (Walsham 1995).

The empirical findings indicate that packaged software upgrade timing is determined by business interests, business calendar, on-going and future development projects, and the vendor. Business interests define the basis and large scale time-frame for IS upgrade timing. It includes the potential business value of the upgrade, minimizing hindrance of the upgrade for business, and the organization's readiness for upgrading. Business calendar defines the possible annual time slots for upgrades through the recurring events in the business, such as high and low seasons of sales, reporting and closing the accounts in the end of the year, production seasons and holiday seasons, etc. IT renewal is timed in a way that it does not overlap with other internal development projects unless they can be combined with joint benefits. IT strategy provides general guidelines for upgrade timing. The system vendor has an influence on the upgrade timing through the provision of new versions, the maturity the new versions and their release timing. These factors define how fast the user organization can upgrade their systems. The vendor also influences on upgrade timing by defining the end-of-support-date for older versions. This pushes the organizations to upgrade faster than needed from business perspective.

This paper was co-authored with Samuli Pekkola. The background literature analysis, data collection and data analysis were conducted by Irja Kankaanpää.

4.5 Paper 5: IS Evolution Benefit Assessment – Challenges with Economic Investment Criteria

Kankaanpää I., Koskinen J., Tilus T., Sivula H., Lintinen H. and Ahonen J. J. (2007) IS Evolution Benefit Assessment - Challenges with Economic Investment Criteria. In W. Abramowicz and H. C. Mayr (Eds.) Technologies for Business Information Systems (pp. 183-191). Netherlands: Springer.

The fifth paper studies the challenges related to the use of economic assessment criteria in IS evolution benefit evaluation. The objective was to chart the available economical methods for IT renewal evaluation and to analyze their suitability for this purpose based on the experiences and findings in related literature. Theoretical research approach was used for investigating the benefit assessment of IS evolution investments because an overview of the research topic was needed. A literature review of the IT benefit evaluation literature was conducted. The concepts of traditional, economic investment evaluation were examined as basis.

Eight traditional investment evaluation criteria were analyzed: net present value (NPV), internal rate of return (IRR), profitability index, payback period, discounted payback period, average accounting return (AAR) (Ross, Westerfield and Jordan 1998, p. 23), return on investment (ROI) method and real options (Curley 2004, p. 139). Their suitability in IS evolution benefit evaluation was assessed in the light of literature findings. The findings indicate that NPV and ROI are be the most appropriate for IS evolution benefit assessment. They may not be sufficient if used alone and hence should be complemented with other methods, including average accounting return (AAR), internal rate of return (IRR) and payback period (see Ross et al. 1998, p. 23). The high uncertainty typically related to IT projects, project timing and risks are addressed the best by real options method (Curley 2004, p. 139). Four major challenges were identified in the IS evolution investment evaluation. The first challenge is the selection of suitable benefit assessment methods. Each method has its strengths and weaknesses that define the exact area where it is applicable. The second challenge is to collect the input data needed to carry out the calculations in accordance with the selected method (Verhoef 2002). Without proper data even the best method can only produce sophisticated guesses. The third challenge related to the use of economic evaluation criteria is that they are unable to capture intangible and unquantifiable benefits common to IT investments (Remenyi et al. 2000). The fourth challenge is the identification of the potential evolution options and comparing them with consistent manner, i.e. being able to identify a method or a set of methods that can capture the benefits of all the evolution alternatives in order to enable their reliable comparison. The theoretical proposition of this paper is that the IS evaluation method should be selected according to the organizational skills, available resources, available data, investment characteristics and whether there is a need to compare different evolution alternatives. One-method-fits-all does not exist in IS evolution evaluation. This paper approaches the IS evolution benefit assessment problem from theoretical perspective. It was conducted in a form of literature analysis. The purpose of the paper was to provide theoretical basis for further empirical investigation that is reported in paper 6.

This paper was planned by Irja Kankaanpää, Jussi Koskinen, Tero Tilus, Henna Sivula, Heikki Lintinen and Jarmo J. Ahonen. The literature survey for the theoretical background, the analysis and writing the paper was carried out by Irja Kankaanpää. Jussi Koskinen, Tero Tilus, Henna Sivula, Heikki Lintinen and Jarmo J. Ahonen reviewed the paper and provided with valuable commentaries and suggestions for improvement.

4.6 Paper 6: ISEBA – A Framework for IS Evolution Benefit Assessment

Kankaanpää I., Sivula H., Ahonen J. J., Tilus T., Koskinen J. and Juutilainen P. (2005) ISEBA – A Framework for IS Evolution Benefit Assessment. In D. Remenyi (Ed.) the Proceedings of the 12th European Conference on Information Technology Evaluation (ECITE 2005) (pp. 255-264). UK: Academic Conferences International.

The sixth paper approaches IT artefact renewal from benefit assessment perspective. The objective was to address the challenge of IT renewal benefit assessment by developing a framework to assist the selection of a suitable benefit evaluation method. The paper adopted constructive research approach. Constructive research builds a new artefact based on existing research and need for improvement, be it technical or organizational (Järvinen 1999, p. 59). The purpose was to construct a tool that would promote a systematic evaluation method selection process. The ISEBA framework consists of a process description which is supported by a requirement-method-matrix, input-data descriptions, description of resources and skills, and guidelines for result interpretation. These latter ones are merely referred in the paper, with the exception of requirement-method-matrix of which an example is given. The design itself was a long term search process through literature and existing research. The experiences from the application of ISEBA framework indicate that the availability of resources and the capabilities for input-data collection influence the selection of the evaluation method. Familiar methods are more easily accepted than unfamiliar ones. In company cases, the selected benefit assessment methods included user satisfaction, system's suitability to organization's business processes, technical quality and business value of the system. Financial benefit assessment was considered interesting but was not conducted due to missing data for the calculations of costs and benefits.

This paper was co-authored with Henna Sivula, Jarmo J. Ahonen, Tero Tilus, Jussi Koskinen and Päivi Juutilainen. Irja Kankaanpää conducted the literature analysis, framework and method-matrix development and writing the paper. Tero Tilus and Jussi Koskinen carried out the visual design of the framework structure (Figure 1 in paper 5). Case 1 was conducted by Tero Tilus, Henna Sivula and Heikki Lintinen. Case 2 was conducted together with Henna Sivula. All authors participated in reviewing and commenting the paper.

4.7 The interrelationship between the papers

The dissertation includes six papers. Each one approaches the research objective from a different angle (see FIGURE 15). The first paper defines the research context by establishing the IS life cycle from organizational point of view based on the work system life cycle (see Alter 2008a). It presents conceptual and theoretical analysis of IS as a living entity that continuously evolves and adapts to environmental changes through the repetitive life cycle stages. It demonstrates that an IS as a socio-technical system shares the characteristics of living entities and, consequently, it justifies the use of life cycle theory in IS context. The second paper complements the aforementioned study on IS life cycle with empirical findings. It reveals that in practice IT life cycle is perceived in three ways: as a linear sequence of stages, as life time of a system or as continuously evolving system that it integrated to the processes it supports depending on the view the perceiver has towards the IS (technical or socio-technical view). Papers 1 and 2 seek answers to RQ1 and simultaneously serve as a foundation for the rest of the included papers. Paper 2 also provides answer to RQ2-1 as it includes reasons for IT artefact renewal.

The third paper presents results of an empirical study of factors that trigger IT artefact renewal in form of modernization and replacement. This paper complements the IS life cycle framework by defining the driving forces that initiate a new cycle. The third paper contributes to RQ2 by providing an answer to its sub-question RQ2-1.

The fourth paper studies the factors influencing the timing of IT artefact renewal and its possible time frames in organizations. It contributes to RQ1 by answering RQ1-2. Paper 4 also contributes to the question about renewal triggers (RQ1-1) due to finding that most renewal triggers play important part in defining the renewal timing.

The fifth paper approaches the IT artefact renewal from benefit evaluation perspective. The suitability and challenges of economic benefit assessment methods are analyzed by means of literature review. The findings indicate that the application of traditional economic benefit assessment criteria in IT artefact renewal context is challenging and possibly misleading.

The sixth paper studies the IT artefact benefit assessment in practice. It forms the constructive part of the thesis by introducing a framework for IS evolution benefit assessment (ISEBA) to support the method selection for IT artefact renewal evaluation.

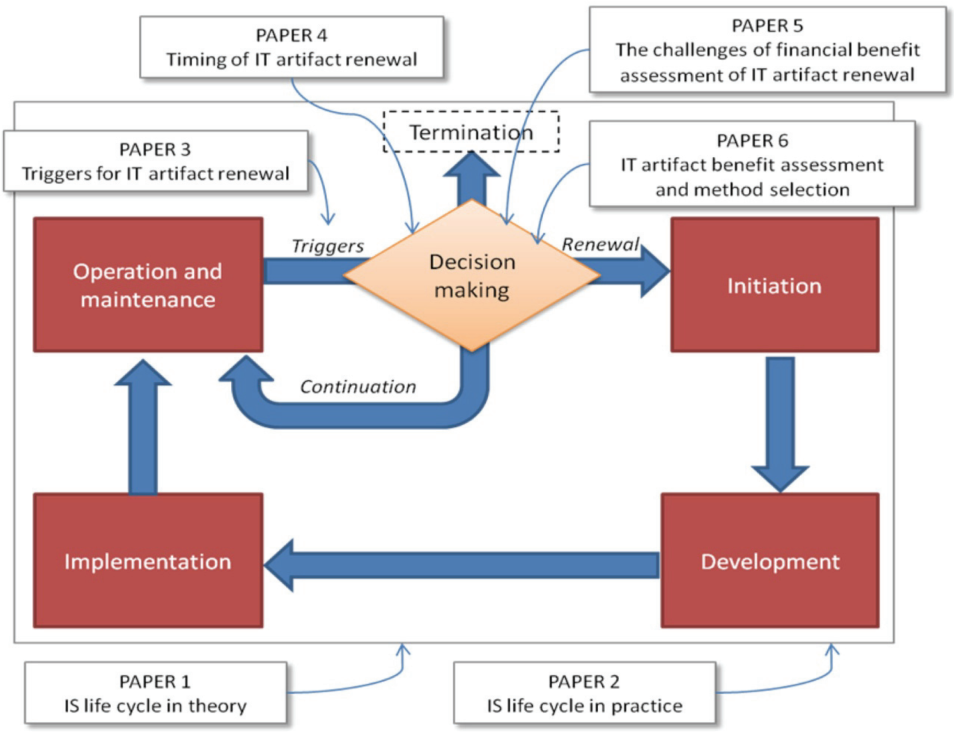

FIGURE 15 The research construct with the related articles.

In summary, papers 1 and 2 define the research context. Papers 3, 4, 5 and 6 contribute to the determinants of IS renewal initiation. The correspondence of the included papers to the research questions is summarized in TABLE 3.

TABLE 3 The included papers and the research questions

	RQ1	**RQ1-1**	**RQ2**	**RQ2-1**	**RQ2-2**	**RQ2-3**
Paper 1	X	X				
Paper 2	X	X	X	X		
Paper 3			X	X		
Paper 4		X	X	X	X	
Paper 5			X			X
Paper 6			X			X

5 DISCUSSION OF THE RESULTS

The main research questions addressed in this dissertation were: *What is the life cycle of an IS from organizational point of view?* and *What are the determinants for IT artefact renewal in organizations?* These questions are connected to each other because a new cycle in IS life begins through its renewal. The following sections provide answers to the research questions.

5.1 IS life cycle from organizational perspective

The understanding of IS life cycle is a precondition for managing it in organizations. The current life cycle models do not support management function or the organization's perspective as they do not engage IT with organizational change (McGann and Lyytinen 2008): Development models approach IS life cycle as a development project; implementation oriented life cycle models treat the IS life cycle as an implementation project. However, when viewed from organizational perspective, development and implementation are both events in a sequence of events the IS undergoes during its existence at the service of the organization. The organization that uses the IS views it from a wider perspective, as part of the organizational processes and tasks.

The research question *What is the life cycle of an IS from organizational point of view?* seeks perceptions on IS life cycle and the nature of IS life cycle in organizations. The use of life cycle metaphor in IS context is justified by attesting that IS, as a socio-technical system, exhibits the general life characteristics. IS life cycle is perceived in organizations in three different ways: 1) as life time of the system (defined in years), 2) as life span of the system (defined as a linear sequence of stages) and 3) as a continuously evolving part of processes and tasks it supports (defined as a cycle). The interpretation of these perceptions reveal a dual view towards information systems: On one hand they are perceived as technical systems while on the other hand they are perceived as socio-technical systems. Technical view corresponds to IT artefact definition by Orlikowski and Iacono (2001, p. 121) as

> "bundles of material and cultural properties packaged in some socially recognizable form such as hardware and/or software".

The socio-technical view, in turn, corresponds to Alter's (2008a) work system and to Leavitt's (1965) socio-technical model.

The life time and life span are related to the life cycle of IT artefact. The life cycle of a technical system is perceived as a life stages from cradle to grave, consisting of identification of need, decision to start, requirements specification, acquisition or development project, implementation, use and maintenance, further development (modifications and upgrades) close-down phases. This approach is similar to Esteves and Pastor's (1999) ERP system life cycle model and also correlates to traditional ISD process (Avison and Fitzgerald 2008). The life cycle of IT artefact is also evaluated on the basis of system's life time in organizational use (FIGURE 16). In related literature this has correlation to software lifetime (Tamai and Torimitsu 1992) and useful life (Tan and Mookerjee 2005). Interestingly, the maximum (30 years) and minimum (1 year) of system life time correspond to those of twenty years ago (Tamai and Torimitsu 1992).

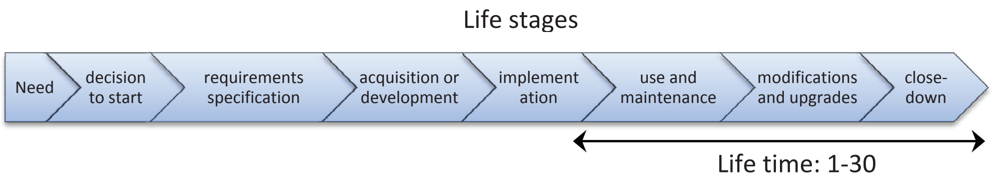

FIGURE 16 Linear life stages and life time of IT artefact

The life cycle of a socio-technical system is perceived as a continuously evolving system, similar to WSLC model (Alter 2008a). Its life cycle phases were not explicitly defined by the CIOs yet it was considered as a never-ending cycle where the IS evolution follows closely the life cycle of organization's processes and tasks. The IS is integrated in organizational processes and tasks in such depth that they form an inseparable entity. Hence, it is called *process-integrated* life cycle. In FIGURE 17, the WSLC model has been adapted to depict the process-integrated IS life cycle view. As an IS is a sum of processes and activities, participants, information and technology, as suggested by Alter (2008a), this adapted model shows how these four elements share the same life cycle and, as consequence, change together.

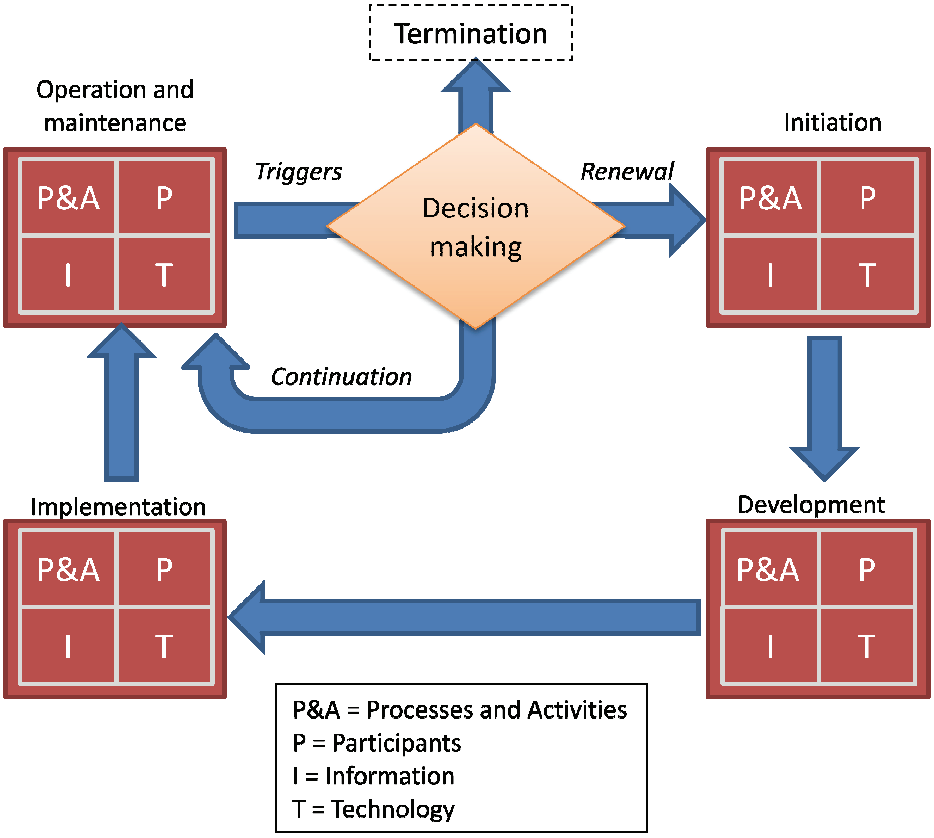

FIGURE 17 Process-integrated IS life cycle

Hence, the *answer to RQ1* and its sub-question *RQ1-1* is: IS life cycle is the sum of perceptions one has towards IS itself and its life. The definition of IS life cycle depends on whether it is perceived as an IT artefact or a work system, or as technical or socio-technical system respectively. IT artefact life cycle is a linear representation of life stages or life time from beginning to an end. IS life cycle is a cycle that enables its continuous change through recurring stages. These results are summarized in TABLE 4.

TABLE 4 The IS and IT artefact life cycle views

IS view	**IS definition**	**IS life cycle view**	**Similar views in literature**
Technical	IT artefact	Linear, lifetime	Software lifetime (Tamai and Torimitsu 1992), useful life (Tan and Mookerjee 2005)
		Linear, life stages	ERP life cycle (Esteves and Pastor 1999), ISD life cycle (Avison and Fitzgerald 2008)
Socio-technical	Work system	Cyclical, process-integrated	WSLC (Alter 2008a), socio-technical model (Leavitt 1965)

5.2 The determinants of IT artefact renewal

The second research question "*What are the determinants for IT artefact renewal in organisations?*"addresses the need to justify IT artefact renewal decisions. Such decisions should be based on the rationale of *why* the renewal is needed and *what benefits* does it renewal provide and *when* is the best time to renew. The alternatives for IT artefact evolution include termination of the IT artefact (decision alternative (DA1), renewal of IT artefact (DA2) or continuing maintenance (DA3) (see FIGURE 18). The first alternative terminates the life of the particular IT artefact. The second alternative initiates a new evolutionary cycle through radical change. The third alternative means continuing the use and maintenance and adaptation of IT artefact through incremental adaptation.

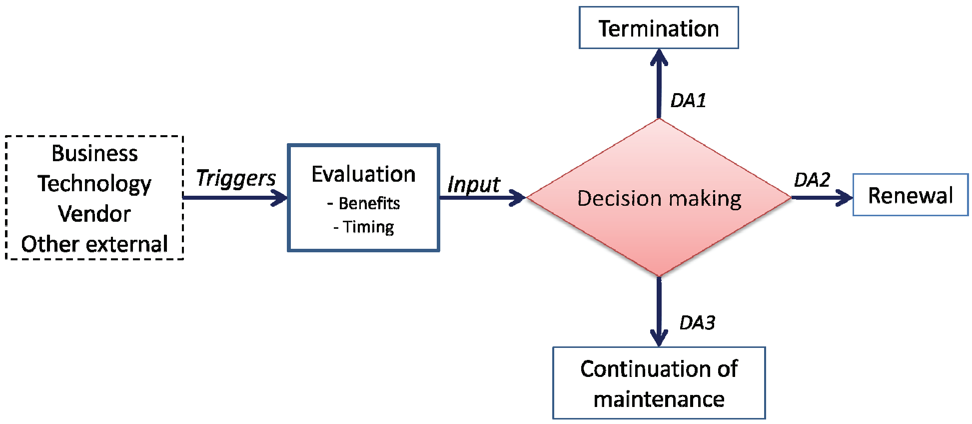

FIGURE 18 Decision making of IT artefact evolution

The second research question was approached from three perspectives, i.e. IT artefact renewal triggers, benefits and timing that constitute the IT artefact renewal determinants (see TABLE 5). Respectively, three sub-questions were formulated.

The first sub-question *What are the triggers for IT artefact renewal?* was examined by means of literature analysis and empirical study. The empirical findings show that IT renewal triggers emerge from three sources, namely technology, business and vendor. Technology related triggers are for IT artefact renewal include the system's old age, obsolete technology and high operation or maintenance costs, incompatibility with other systems and need to unify separate systems and scattered data, system's incompliance with company's business needs. Business related triggers include need for business or process development and compliance with IT strategy. Vendor triggers IT artefact renewal, particularly replacement, by regulating how long it will provide system support for each version. IT artefact renewal decisions were also influenced by external factors such as change of legislation, requirements from business partners and customers and IT trends in field of industry. Vendor is presented separately from other external factors because it is a special case among them. Its triggering influence on IT artefact renewal and particularly on its timing is significantly greater compared to the other external forces.

TABLE 5 Summary of the IT artefact renewal determinants

Determinant	**General**	**Detailed**
Triggers	Technology	• system's old age • obsolete technology • high operation and maintenance costs • incompatibility with other systems • unification of separate systems or scattered data • incompliance with business needs
	Business	• business or business process development • compliance with IT strategy
	External, vendor	• end of system support
	External, other	• statutory changes • requirements from business partners • requirements from customers • IT trends in field of industry
Benefits	Economic benefits	• NPV • ROI
	Intangible benefits	• user satisfaction • system's suitability for business processes • technical quality of the system • business value of the system
Timing	Business interests	• business value of the upgrade • minimizing hindrance for the business • available resources
	Business calendar	• high and low seasons • closing of the accounts and reporting in the end of fiscal year • holiday seasons
	Development projects	• on-going projects • future projects
	IT strategy	• time frame for regular upgrades
	Vendor	• availability of new versions • maturity the versions • version release timing • sunset-date

The benefit evaluation was addressed in the sub-question *How are the IT artefact renewal benefits evaluated?* The answer to this question is: IT artefact renewal benefits are evaluated case by case, depending on the renewal triggers. In the case companies the most effective and applicable evaluation methods were user satisfaction questionnaire and stakeholder interviews. Intangible benefit evaluation methods were preferred over economic criteria. Familiar methods were preferred over unfamiliar methods. The evaluation methods that used minimum amount of resources and that were proven to work are preferred over formalized methods. Traditional economic evaluation criteria, including ROI and NPV, are insufficient and challenging to apply in IT artefact renewal benefit evaluation. The greatest challenges are collection of input data for the calculations in accordance with the selected method, the inability of the economic criteria in capturing the intangible and unquantifiable benefits, and the identification of the potential evolution alternatives and comparing them to each other systematically. The selection of suitable method(s) for IT artefact renewal benefit evaluation was recognized as one of the most serious problems in IT investment benefit evaluation in the literature. Despite of the plentiful supply of methods the IT renewal benefit evaluation is a difficult task. In both ISEBA test-cases the firstly selected evaluation criteria was later modified by removing the criteria the measurement of which would have been resource consuming or otherwise considered difficult to carry out. This boils down to the question of evaluating the evaluation effort itself – how much evaluation is enough and how much benefits the evaluation of the IT investment yields? In the end, the user satisfaction was the major indicator for the need of change in the studied case companies.

The research question *What factors define the IT artefact renewal timing?* was examined by CIO interviews in order to identify conditions that define the upgrade timing. The findings indicate that packaged software upgrade timing is determined by organization’s business interests, business calendar, on-going and future development projects, and the vendor. Business interests include the business value of the upgrade, minimizing hindrance for the business, and the organization’s readiness (available resources) for an upgrade. Business calendar refers to the recurring business activities including high and low seasons, reporting and closing the accounts in the end of fiscal year, and holiday seasons. The business calendar is company and business field specific and strictly defines the possible time slots for upgrades. Upgrade timing is dependent on organization’s IT strategy and project portfolio, i.e. on-going and planned projects. In the studied companies the timing strategy for upgrades was to avoid overlapping with other projects or to combine them with other projects if this was considered beneficial. IT strategy regulates timing by defining a time frame for regular upgrades. Upgrade timing is influenced by system vendor through the availability of new, suitable version, the maturity the new versions and their release timing. Vendor’s sunset-date for older version support pushes customers to upgrade in order to avoid the risks of running unsupported applications.

The findings reveal that majority of the factors that trigger IT artefact renewal also influence the renewal timing. Business interests, e.g. need for process development, are a major trigger and simultaneously define the time frame for the renewal, i.e. when does the IT artefact renewal investment yield the greatest benefits and causes minimum hindrance to the business. Vendor influences both in the decision to renew, e.g. by announcing the end of support for the current version, and the timing by giving the sun-set-date for the support. In general, the business needs of the user-organization and the system vendor define the *urgency* of timing through the amount of pressure they apply on the need to renew. Project portfolio defines how resources are allocated for different projects and through this it sets the *possibilities* and *limitations for timing*. It defines whether IT renewal is possible and when there are sufficiently available resources for it. Also vendor sets possibilities and limitations on the timing through the availability of new versions, maturity of a new version, and version release schedule. These define the limits how fast the user-organization can upgrade their systems if they desire to do so. The on-going and future projects influence on the *timing strategy* of IT artefact renewal, i.e. whether to combine IT renewal project with other on-going projects or to avoid overlapping projects. The timing strategy is also influenced by the expected benefits from the IT renewal. It may be strategically better, for instance, to wait for other organizations to test and adopt certain technology or new version first and learn from their experiences, or to skip versions that do not have suitable features as suggested by Mukherji et al. (2006). As a synthesis of the empirical and literature findings the IT artefact timing determinants define the urgency of timing, the possibilities and limitations of timing and the timing strategy.

5.3 Contributions

The thesis makes the contributions to research and practice by providing novel results concerning conditions influencing IT artefact renewal in organizations. It contributes to both research and practice by increasing the understanding of the IS life cycle form organization's perspective. This is done by explaining the previously obscure areas and specifying the scope of WSLC.

The first result is the revised application area of the WSLC model. The WSLC, as such, models the life cycle of an IS as a socio-technical system. The iterative nature of the WSLC is obtained through the integration of its different elements through processes and activities. Together they form an IS that continuously evolves. If IT artefact is taken out of the WSLC model and examined as an individual element the connection to other elements is cut and, as a consequence, the cyclic continuity is lost. Hence, the WSLC fails to model the life cycle of an IT artefact. This makes a contribution to practice, particularly for IT management and IS life cycle management in organizations. The view one has toward IS has a great significance in communication because it is related to a subjective perception of the IS life cycle, that can be either linear or cyclical. These perceptions of IS and its life cycle can be identified through communication between people. If not communicated clearly room is left for misunderstandings and false assumptions. People apparently speak about the "same" IS but in reality it represents different thing for different people, as it is the case within the CIOs. This emphasizes the significance of openly defining the shared concepts throughout the organization so that everybody speak about the IS and its life cycle with the same name and with the same meaning. The revised life cycle model has practical implications to IT management as it contributes in increasing the understanding of the cyclical nature of an IS and conditions that lead to the start of a new cycle.

The second result is the explanation of the "decision diamond" in the WSLC model. Despite of its many revisions, the WSLC model lacks an explanation for the diamond shaped figure that connects the operation and maintenance phase with the initiation phase. This thesis contributes to theory by providing such explanation: the diamond signifies a decision making point in IS life cycle. It is significant to the overall life cycle because without such decision, planned IS change or IT artefact renewal were not possible. The decision diamond is a part of organizational IT management. It is strategic by nature because it takes into consideration the corporate strategy for maintaining or achieving better business-IT alignment. It may include building strategic partnerships with software vendors – or it may include terminating them. It may initiate a radical change in the organization through IT artefact renewal – or it may not change anything for the time being. Regardless of the *outcome* of the decision, the decision diamond includes three types of *evaluation* that provide input for the actual decision making and support the building of a business case.

The results provide guidelines for IT management when evaluating the IT artefact renewal determinants in order to support the actual IT artefact decision making. Firstly, the need for change should be evaluated. This refers to the triggers that originate from the operation and maintenance phase as the alignment between IS and business is monitored. The questions asked during this evaluation should include:

- Does the pressure for change exceed the critical threshold?
- What kind of action is required - small scale adaptation or radical change?

Secondly, the possible alternatives and related benefits should be evaluated. This includes questions such as:

- What are the expected benefits of different alternatives?
- What is the cost of not doing anything?

Thirdly, the timing of the IT artefact renewal should be evaluated. The questions considering IT artefact renewal timing include:

- When is the optimum time for renewal in relation to organizational forces, external forces and contingencies?
- Is the renewal more beneficial when done sooner or later?

The evaluation determines when and how the renewal will be conducted, or whether is conducted at all. It provides basis for the decision to proceed to the IT artefact renewal initiation phase. The purpose of the evaluation is to ensure that the IT artefact renewal is feasible. If no feasible alternative is found there are two options: to return to continue maintenance or to terminate the IT artefact. The evaluation considers also the potential implications of IT artefact renewal with respect to the remaining IS elements, i.e. processes and tasks, participants and information.

The third result is that the renewed WSLC model incorporates the IT artefact renewal determinants. The renewal triggers are related to the operation and maintenance phase during which they emerge. All the renewal determinants, i.e. triggers, benefits and timing, are inputs to the IT artefact renewal decision making. Triggers are results of the pressure for change exceeding the critical threshold during operation and maintenance phase. WSLC model, as Alter (2008a) describes it, does not include triggers. Instead it includes monitoring the work system performance (Alter 2002) or the alignments between IS and the needs of the work system (Alter 2003). The findings of this thesis indicate however that the plain monitoring is not sufficient enough. Instead, the monitoring results have to be evaluated. If there is a clear indication for a need for change then that triggers the decision making procedure. Besides this, the triggers are important in the decision diamond. They state the reason *why* change is needed and hence provide a starting point for the definition of the IT artefact renewal goals. The goals, in turn, cascade forward as they form the foundation for the definition of the benefit evaluation criteria are derived.

The fourth result is the expansion of the scope of Khoo and Robey's (2007) model on ERP upgrade decision. The expansion is based on the following statements: 1) the available time frame is a significant contingency factor in IT renewal decision making, 2) benefits influence the IT artefact renewal decision, 3) legacy system renewal is significant besides packaged system upgrade, and 4) there are differences in the factors that trigger modernization and replacement. Modernization is more business development driven while replacement decisions are typically vendor driven.

The fifth result is that the IT renewal triggers and factors influencing the renewal timing are similar. Business interests and vendor influence the IT artefact renewal timing as they trigger the renewal need. They define the urgency of timing. On-going and future projects influencing the timing correspond directly to the available resources: the more projects, the less available resources. The project portfolio therefore defines the timing strategy with respect to other projects. If synergy benefits were to be expected IT renewal is likely to be combined with business process development projects. Otherwise simultaneous projects with IT renewal are avoided. The company's annual business calendar has a major influence on the renewal timing. It strictly defines the time-periods during which the renewal cannot be done.

These results provide theoretical contribution because they complement the WSLC model and increase the general understanding. The results provide practical contribution because they explain the determinants of IT artefact renewal and provide guidelines for IT artefact decision making for IT management.

6 CONCLUSIONS, LIMITATIONS AND FUTURE RESEARCH

This chapter provides the conclusions and limitations of the study followed by suggested directions for future research and reflections of conducting the research.

6.1 Conclusions

This thesis states that the life cycle of IS, as a socio-technical system, is cyclical rather than linear. Within this cyclic view, the IS is renewed at certain intervals in order to stay aligned with organizational needs. The renewal of the IT artefact is labor-intensive and costly effort with wide-range influences in organizations. Consequently, the related decisions have to be well justified. Three types of determinants influencing the IT artefact renewal decisions were identified and studied, namely renewal triggers, benefits and timing. Renewal triggers indicate a need for change including its motive and criticality. Benefit evaluation aims at ensuring that the renewal effort yields benefits and is worth of investing organizational resources. Timing is related to identifying a suitable time period for carrying out the IT artefact renewal. Together these three aspects determine the IT artefact renewal.

The IT artefact renewal determinants are interrelated and partly overlapping. In the following the interrelations are briefly described with examples. *Triggers – benefits:* There is a need for IT artefact change which is expected to yield benefits. *Benefits – triggers:* A benefit can be a trigger for change in form of opportunity. For instance company's good economic situation can enable beneficial IT renewal without actual need-based trigger. *Triggers – timing* relation describes how fast the perceived has to be reacted upon, how critical the need for change is. For instance end of vendor support is a trigger that may have high urgency level since action is needed within the timeframe set by the vendor. *Timing – triggers:* Timing related factors can act as triggers when organizational time frames may require IT artefact renewal earlier (or later) than the need-based triggers would indicate. For instance company's IT strategy or vendor's release schedule may dictate IT renewal timing. *Benefits – timing:* The realization of IT artefact benefits depend on their timing. For instance, an IT artefact renewal that is expected to benefit the organization by increasing the IT artefact's suitability to business processes is carried out in the middle of the high sales season. As a result, business is hindered and the realized benefits are lower than expected. *Timing – benefits:* Timing opportunities, for instance in form of joint projects where IT renewal is combined with organizational development project, may yield synergy benefits. The interrelation of the IT artefact renewal determinants is depicted in FIGURE 19.

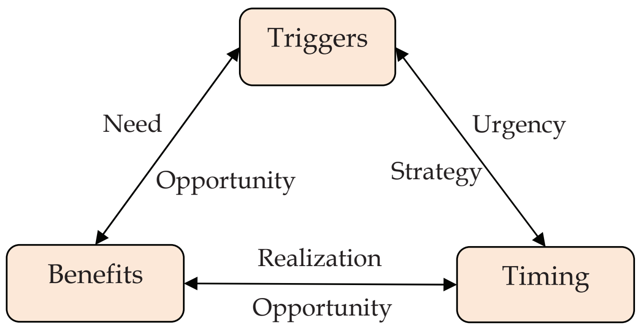

FIGURE 19 The interrelation of the IT artefact renewal determinants

The findings of this thesis coincide with the suggestion of Alter's (2008a) twofold definition of IS that is based on the emphasis on either social concerns or technical concerns. IS is studied from various perspectives and consequently, and not surprisingly, there is a great variety of definitions for IS (Alter 2008a). However, the phenomenon of multi-defined IS does not only occur in the research world. Also the practitioners' community is lacking a uniform definition of IS. There is a need for a holistic IS life cycle that can help both practitioners and researchers to position the IS evolution and renewal mechanics in a larger framework and in a pattern of continuity and, by doing so, assisting the IS life cycle management in organizations.

6.2 Limitations

This study has certain limitations that should be considered while interpreting the results. Firstly, the thesis approaches the IS life cycle and evolution from the organization's perspective. Vendor's perspective is not included (except in the light of literature) and hence the thesis gives a one-sided picture of the studied phenomenon. Also, only one to two persons were interviewed in most organizations and hence the results represent the personal experiences and opinions of the interviewees rather than the organization's. However, the data naturally reflects the organizational context and in the end, the results are interviewee's personal perceptions through the particular organizational lenses. The results might have been different if data collection was inquired from multiple persons in the organizations. Nevertheless, one of the aims of the thesis was to collect CIO perceptions and in that it succeeded.

Secondly, the conduct of the research has some limitations. The empirical research was conducted mainly by the means of interview. The use of additional research methods would have provided better methodological triangulation. The results would have been possibly improved by either using additional quantitative methods in order to acquire data from a larger sample, or by using in-depth case analysis of one or two cases in order to build a deeper case description and analysis. However, as the research topic is new research methods have to be able to provide deep insights and understanding of it. Hence, the use of qualitative, deep-drilling methods, such as case study method, and interview, as data acquisition technique, are justified.

The selection method for research targets also has its limitations. The interviewees were given the freedom to choose the renewal cases from their organization's past for the interviews. A possible danger related to this approach is that the interviewees chose only successful cases for the interview and left out poorly conducted projects. This may have skewed the results and given more positive image of the IT renewal projects than a random sample would have.

Thirdly, the incomplete publication of ISEBA framework limits the replication of the research. The requirement-method-matrix, which is a central supporting element in the selection of benefit evaluation methods, is only partially presented in the fifth paper. ISEBA framework can also be subjected to critics because its intention was to refine the earlier models for method selection (Andresen 2002 and Farbey et al. 1992) yet there was not sufficiently data available of Andresen's model, for instance, to enable its thorough investigation which would have been a prerequisite for systematic improvements and transfer in IT renewal context. Also, the research process in the constructive part of the thesis was not as rigorously documented nor tested as it could have been and it is based on rather old references.

Fourthly, the target of study was geographically limited to Finland. The national influence on the organizational culture sets limitations to the interpretation of the results. This should be taken into consideration when transferring the results in other countries or making generalizations of them.

6.3 Directions for future research

The most promising life cycle model found in the literature analysis was Alter's (2008a) WSLC model. The IS work system consists of participants, processes and activities, information and technology, the life cycle of which consists of a continuous flow of operation and maintenance, initiation, development, and implementation. As the life cycle model was broken down and work system elements were assigned their own life cycle the focusing of the research on the life cycle of the technical element (IT artefact) was made possible. At the same time questions rose considering the life cycle of the remaining IS elements. Are the WSLC phases suitable for describing information life cycle, participant life cycle or the life cycle of processes and activities? Does the WSLC model apply to its elements individually? These questions require further research.
Viewed from organizational perspective, the strength of the WSLC model (Alter 2008a) is its cyclic nature. Its weakness is the close semblance of the traditional ISD process with respect to its terminology and definition of phases. This gives possibilities for future research in adapting the WSLC model to fit better the organizational domain, for instance the terminology should be migrated from technical to organizational. A preliminary suggestion for new phase names is: Use and maintenance, initiation, acquisition or development, implementation (see FIGURE 19 and TABLE 6). Yet, a question remains: would this kind of naming describe the life cycle of information, participants or processes and activities? Is a work system participant, for instance, initiated, developed, implemented, and operated and maintained? More research is needed to define the life cycle of these elements of IS.

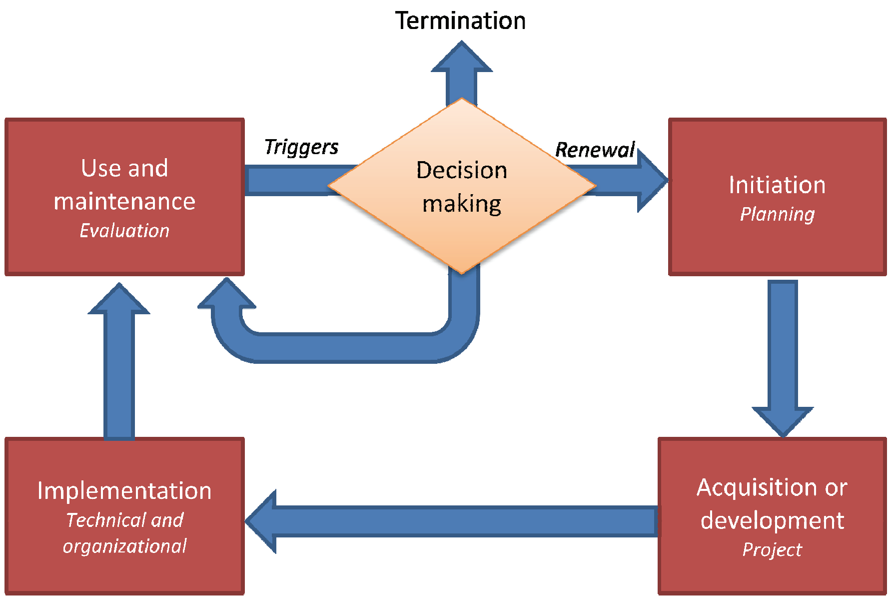

FIGURE 20 The ISLC life cycle

TABLE 6 The revised names of the IS life cycle model

WSLC model (Alter 2008a)	Revised model (ISLC)	Reason for revision
Operation and maintenance	Use and maintenance	IS is a live through its use.
Initiation	Initiation	-
Development	Acquisition or development	Development and acquisition are the most dominant ways of IT artefact renewal.
Implementation	Implementation	-

Change in any of the IS elements influences the remaining elements because they are all connected. As a future research direction the study the renewal of other IS components and the relationships between the changes in different components is suggested. McGann and Lyytinen (2008) have done work on this already from user improvisation perspective.

The thesis fills the gap in the current research concerning the lack of research on IT renewal in general and also lack of studies based on CIO interviews recognized already by DeSanctis (2003). Further research is needed to explore the relationship between the technical and socio-technical view of IS and their implications on IS life cycle management practices in companies. Engaging vendor's perspective would provide a new dimension in understanding the IS life cycle and IT artefact renewal. Also, a long-term research on the life events of an IS is suggested to provide empirical evidence on the IS life cycle. Also, it would be beneficial to study the differences of life cycles between different types of systems in more detail and by empirical means.

This thesis gives preliminary indication that there are differences on how timing in defined at industries. The differences in the timing determinants in different fields of industry are worth of studying. This would provide more reliable guidelines across the fields of industry on the timing of IS upgrades for both practitioners and researchers.

6.4 Reflections

The research topic was challenging, particularly concerning the benefit evaluation. As long as IS is seen as a merely technical system its costs and benefits are impossible to show because they manifest themselves through the tasks and processes. The key is to understand IS as part of the organizational processes and tasks. This, on the other hand, is a cause for the challenges of the IT related benefit evaluation. The benefits manifest in such unpredictable forms and ways that pinning them down from the processes can be very demanding and resource consuming process itself. Hence, a golden mid-way should be found. Enough evaluation should be done in order to provide a best possible estimate with the available resources. This understanding has to reach the business management level, not only IT management department. The CIO interviews reveal that most of the CIOs suffer from the fact that the business management does not understand the necessity of commitment to IT artefact renewal. They want to wash their hands and leave the IT to the IT personnel. Nevertheless, the IT is for the organization and hence its functioning requires interaction with the organization.

The research was satisfying to conduct. The selection of the organization's perspective towards a topic that is commonly considered as belonging under ISD domain appeared extremely rewarding. The collected material was abundant and provided insight to the CIO's perceptions. In this thesis only a fraction of the whole material was reported. Plenty remains for future analysis. The decision to make the thesis as a collection of papers provided an opportunity for continuous refinement of the research topic and research skills. The step-by-step advancing technique enabled the redirection of the research based on the results of the precedent papers and findings. Also, the personal learning process of the researcher is identifiable through the papers published during the five year period of the whole dissertation work. This is visible particularly in the nature of the papers. Their focus has shifted from the hard, economical measures to philosophical contemplation of a socio-technical IS.

YHTEENVETO (FINNISH SUMMARY)

Tässä väitöskirjassa tutkitaan tietojärjestelmän (TJ) elinkaarta sekä tekijöitä, jotka vaikuttavat informaatioteknologia-artefaktin (ITA) uudistamiseen organisaatioissa. TJ on määritelty sosio-tekniseksi systeemiksi ja ITA sen tekniseksi komponentiksi. Organisaatiolla tarkoitetaan mitä tahansa organisoitunutta ryhmää (yritys tai julkishallinnollinen organisaatio), joka käyttää tietojärjestelmiä toimintansa tueksi. Tutkimuksessa on otettu siis organisatorisen käyttäjän näkökulma tarkasteltavaan kohteeseen.

Syklisen näkökulman mukaan TJ:ää on uudistettava tietyin väliajoin, jotta se pystyisi tukemaan organisaatiota ja sen tietotarpeita muuttuvassa liiketoimintaympäristössä. ITA:n uudistamishankkeet ovat usein työläitä ja kalliita ja niillä on laaja-alainen vaikutus organisaatiossa. Tästä johtuen uudistamispäätöksien tulee olla huolellisesti perusteltuja.

Tässä tutkimuksessa TJ:n muutosta tarkastellaan ITA:n näkökulmasta sitomalla se TJ:n elinkaareen. Elinkaari koostuu joukosta toistuvia vaiheita, joiden tuloksena ”eläjä” kehittyy ja uudistuu. Tutkimuksen teoreettisena linssinä käytettiin Alterin (2008a) työsysteemin elinkaarta, jonka yksi elinsykli (life cycle) koostuu seuraavista, toistuvista vaiheista: aloitus, kehitys, käyttöönotto ja ylläpito. Syklisen näkökulman mukaan TJ:ää on uudistettava tietyin väliajoin, jotta se pystyisi tukemaan organisaatiota ja sen tietotarpeita muuttuvassa liiketoimintaympäristössä. TJ muodostuu sosiaalisista ja teknisistä elementeistä. Epätasapaino elementtien välillä luo muutospainetta tasapainon uudelleensaavuttamiseksi. Kun yksi elementti muuttuu, muut elementit sopeutuvat muutokseen ja näin ollen niiden muodostama kokonaisuus muuttuu. ITA:n uudistaminen on yksi tapa, jonka kautta TJ muuttuu. ITA:n uudistamishankkeet ovat usein työläitä ja kalliita ja niillä on laaja-alainen vaikutus organisaatiossa. Tästä johtuen uudistamispäätöksien tulee olla huolellisesti perusteltuja.

Tutkimuksessa tarkastellaan syklisen elinkaarimallin soveltuvuutta organisaatioihin sekä tekijöitä, jotka vaikuttavat uuden TJ:n elinsyklin käynnistymiseen ITA:n uudistamisen kautta. Päätutkimuskysymykset ovat: *TK1: Mikä on tietojärjestelmän elinkaari organisatorisesta näkökulmasta katsottuna?* ja *TK2: Mitkä ovat IT-artifaktin uudistamisen määräävät tekijät organisaatiossa?*

Tutkimuksen empiirinen osa toteutettiin laadullisia tutkimusmenetelmiä käyttäen. Tutkimuksen kohderyhmänä ovat Suomessa toimivat yritykset sekä julkiset organisaatiot. Aineisto koostuu yritysten tietohallintojohtajien sekä ITA:n uudistamishankkeiden avainhenkilöiden haastatteluista.

Tulokset osoittavat, että ITA:n uudistamiseen vaikuttavat kolme tekijää: käynnistävät tekijät (triggers), uudistamisella tavoiteltavat hyödyt (benefits) ja sopiva ajoitus (timing). Tutkimuksen löydökset osoittavat, että TJ:n elinkaari nähdään kahtalaisena organisaatioissa. Sosio-tekniseksi systeemiksi määriteltynä TJ:n elinkaari on syklinen. Tekniseksi systeemiksi määriteltynä TJ:n elinkaari on lineaarinen.

REFERENCES

Adolph, W.S. (1996) Cash Cow in the Tar Pit: Reengineering a Legacy System. IEEE Software, 13 (3), 41-47.

Ahmad, A., Lyytinen, K. and Newman, M. (2011). The evolution of process models in IS research: from a punctuated social process model to a socio-technical process model. In V. Tuunainen, J. Nandhakumar, M. Rossi and W. Soliman (Eds.) 19th European Conference on Information Systems (pp. 1993-2004). Helsinki. http://aisel.aisnet.org/ecis2011/161 (retrieved 30.8.2011)

Alter, S. (2001) Which Life Cycle – Work System, Information System, or Software? Communications of the Associacion for Information Systems, 7 (17), 1-54.

Alter, S. (2002) The work system method for understanding information systems and information systems research. Communications of the Associacion for Information Systems, 9 (6), 90-104.

Alter, S. (2003) 18 Reasons Why IT-Reliant Work Systems Should Replace "The IT Artefact" as the Core Subject Matter of the IS Field. Communications of the Associacion for Information Systems, 12 (23), 366-395.

Alter, S. (2008a) Defining information systems as work systems: implications for the IS field. European Journal of Information Systems 17, 448–469.

Alter, S. (2008b) Service system fundamentals: work system, value chain, and life cycle. IBM Systems Journal 47(1), 71-85.

Andresen, J. L. (2002). How to Select an IT Evaluation Method – In the Context of Construction. In Proceedings of CIB w78 Conference, Denmark: Århus School of Architecture. http://itc.scix.net/cgi-bin/works/Show?w78-2002-16 (retrieved 20.5.2005).

Antikainen, J. and Pekkola, S. (2009) Factors influencing the alignment of SOA development with business objectives. In S. Newell, E.A. Whitley, N. Pouloudi, J. Wareham and L. Mathiassen (Eds.) Proceedings of the 17th European Conference on Information Systems (pp. 2579-2590), Verona, Italy.

Avison, D. and Fitzgerald, G. (2008) Information systems development, 4th edition. UK: McGraw-Hill

Beatty, R.C. and Williams, C.D. (2006) ERP II: best practices for successfully implementing an ERP upgrade. Communications of the Association for Information Systems, 49 (3), 105-109.

Benbasat, I. and Zmud, R.W. (2003) The Identity Crisis Within the IS Discipline: Defining and Communications the Discipline's Core Properties. MIS Quarterly, 27 (2), 183-194.

Bennett, K. (1994) Legacy systems: Coping with success. IEEE Software, 12 (1), 19-23.

Bennett, K. and Rajlich, V. (2000) Software maintenance and evolution: a roadmap. In Proceedings of the Conference on The Future of Software Engineering (ICSE '00) (pp. 73-87). New York: ACM.

Bennett, P. and Owens, I. (2002) Evolutionary Ecology of Birds: life histories, mating systems, and extinction. UK: Oxford University Press.

Boehm, B. W. (1988) A Spiral Model of Software Development and Enhancement. Computer, 21 (5), 61-72.

Bogin, B. and H. Smith (1998) Evolution of the human life cycle. American Journal of Human Biology, 8 (6), 703-716.

Brynjolfsson E. (1993) The productivity paradox of information technology. Communications of the ACM, 36 (12), 66-77. http://doi.acm.org/10.1145/163298.163309 (retrieved 5.9.2011)

Campbell, J. A. (2002) Real options analysis of the timing of IS investment decision. Information & Management, 39, 337-334.

Campbell, N. A. and Reese, J. B. (2002). Biology. San Francisco: Benjamin Cummings.

Chang, S., Yen, D.C., Huang, S. and Hung, P. (2008) An ERP System Life Cycle-Wide Management and Support Framework for Small- and Medium-Sized Companies. Communications of the Association for Information Systems, (22) 15, 275-294.

Chapin, N. (2000). Software maintenance types–a fresh view. In the International Conference on Software Maintenance (pp. 247–252). Los Alamitos: IEEE Computer Society.

Conolly, D. (2000). A little history of the Worl Wide Web. http://www.w3.org/History.html (retrieved 1.12.2010).

Curley, M. (2004). Managing Information Technology for Business Value, Hillsboro, OR, USA: Intel Press.

Damsgaard J. and Karlsbjerg J. (2010). Seven principles for selecting software packages. Communications of the ACM, 53 (8), pp. 63-71.

Darwin, C. (1968). The Origin of Species by Means of Natural Selection or the preservation of favoured races in the struggle for life. J. W. Burrow (Ed.) Middlesex, Great Britain: Penquin Books.

Davenport, T.H. (1998). Putting the Enterprise into the Enterprise System. Harvard Business Review, Jul-Aug, 1-11.

Davis, AM, Bersoff, E.H. and Comer, E.R. (1988). A Strategy for Comparing Alternative Software Development Life Cycle Models. IEEE Transactions on Software Engineering, 14 (10), 1453-1461.

Davis, G. B. and M. H. Olsen (1985). Management Information Systems: Conceptual foundations, structure, and development. Singapore: McGraw-Hill.

Dekleva, S. (2005). Justifying investments in IT. Journal of Information Technology Management, XVI (3), 1-8.

Denzin, N.K. and Lincoln, Y.S. (2000). Introduction: The Dicipline and Practice of Qualitative Research. In N.K. Denzin and Y.S. Lincoln (Eds.) Handbook of Qualitative Research. California: Sage Publications.

DeSanctis, G. (2003). The Social Life of Information Systems Research: A Response to Benbasat and Zmud's Call for Returning to the IT Artefact. Journal of the Association for Information Systems 4(1), 360-376.

Devaney, R. L. (2007). Mathematical Glossary (On-Line). http://math.bu.edu/DYSYS/explorer/def.html (retrieved 30.11.2007).

Dey, I. (1993). Qualitative data analysis: A user-friendly guide for social scientists. Bristol, UK: Biddless.

Eisenhardt, K.M. and Graebner M.E. (2007). Theory building from cases: Opportunities and challenges. Academy of Management Journal, 50 (1), 25–32.

Esteves, J. and Pastor, J. (1999). An ERP lifecycle-based research agenda. In International Workshop on Enterprise Management Resource and Planning Systems (EMRPS) (pp. 359-371). Venice, Italy.

Farbey, B., Land, F. and Targett, D. (1992). Evaluating Investments in IT. Journal of Information Technology, 7, 109-122.

GAO (1996). Information technology investment: Agencies Can Improve Performance, Reduce Costs and Minimize Risks. United States General Accounting Office, Report to Congressional Requesters, GAO/AIMD-96-64. www.gao.gov/archive/1996/ai96064.pdf (retrieved 24.1.2011).

Gersick, C.J.G. (1991). Revolutionary Change Theories: A Multilevel Exploration of the Punctuated Equilibrium Paradigm. The Academy of Management Review, 16 (1), 10-36. http://www.jstor.org/stable/258605 (retireved 11.1.2011).

Gould, S.J. and Eldredge, N. (1977). Punctuated Equilibria: The Tempo and Mode of Evolution Reconsidered. Paleobiology, 3 (2), 115-151. http://www.jstor.org/stable/2400177 (retrieved 11.1.2011).

Greschler, D. and Mangan, T. (2002). Networking lessons in delivering, Software as a Service' Part I. International Journal of Network Management, 12 (5), 317–321.

Haddara, M. and Päivärinta, T. (2011). Why Benefits Realization from ERP in SMEs Doesn't Seem to Matter? In the 44th Hawaii International Conference on System Sciences (HICSS) (pp. 1-10). IEEE Computer Society Press.

Hanseth, O. and Lyytinen, K. (2004). Theorizing about the Design of Information Infrastructures: Design Kernel Theories and Principles. Sprouts: Working Papers on Information Systems, 4(12). http://sprouts.aisnet.org/4-12 (retrieved 29.11.2011).

Hirschheim, R.A., Klein, M. and Lyytinen, K. (1995). Information systems development and data modelling: Conceptual and philosophical foundations. Cambridge: Cambridge University Press.

Hirsjärvi S., Remes P. and Sajavaaara, P. (2010) Tutki ja kirjoita (15th -16th edition). Helsinki: Tammi.

Holton G. A. (2004). Defining Risk. Financial Analysts Journal, 60 (6), 19-25. http://www.riskexpertise.com/papers/risk.pdf (retrieved 22.3.2011)

Iivari, J. (2007). A Paradigmatic Analysis of Information Systems As a Design Science. Scandinavian Journal of Information Systems, 19 (2), 39-64.

Irani, Z. and Love, P.E.D. (2002). Developing a frame of reference for ex-ante IT/IS investment evaluation. European Journal of Information Systems, 11 (nro), 74–82.

Järvinen, P. (1999). On Reserach Methods. Tampere: Opinpaja.

Jick, T. D. (1979). Mixing Qualitative and Quantitative Methods: Triangulation in Action, Administrative Science Quarterly, 24 (4), 602-611. http://www.jstor.org/stable/2392366 (retrieved 31.01.2011).

Kazman, R. R., Nord, R. L. and Klein M.H. (2003). A Life-Cycle View of Architecture Analysis and Design Methods. Technical Note (CMU/SEI-2003-TN-026).

Keen, P. G. W. (1981). Information Systems and Organizational Change Communications of the ACM, 24 (1), 24-33.

Khoo, H. M. and Robey, D. (2007). Deciding to upgrade packaged software: a comparative case study of motives, contingencies and dependencies. European Journal of Information Systems, 16, 555-567.

Kirk, J. and Miller, M. L. (1986). Reliability and validity in qualitative research. Qualitative research methods series 1. USA: Sage University Paper.

Kremers, M. and van Dissel, H. (2000). Enterprise resource planning: ERP system migrations. Communications of ACM 43 (4), 53-56.

Leavitt, H. J. (1965) Applied Organizational Change in Industry: Structural, Technologucal and Humanistic Approaches. In J. G. March (Ed.) Handbook of Organizations (1144-1170). Chicago: Rand McNally & Company.

Lee, A. S. and Baskerville, R. L. (2003). Generalizing Generalizability in Information Systems Research. Information Systems Research, INFORMS, 14 (3), 221-243.

Lehman, M. M. and Ramil, J. F. (2003). Software evolution—Background, theory, practice. Information Processing Letters, 88 (1-2), 33–44

Lewin, K. (1952). Group Decision and Social Change. In G. E. Swanson, T. M. Newcomb, E. L. and Hartley (Eds.) Readings in Social Psychology, revised edition (pp. 459 – 473). New York: Holt.

Lyytinen, K. and Newman, M. (2008). Explaining information systems change: a punctuated socio-technical change model. European Journal of Information Systems, 17 (6), 589-613.

Mahmood, M. A. and Mann, G. J. (1993). Measuring the organizational impact of information technology investment: an exploratory study. Journal of Management Information Systems, 10 (1), 97-122.

Markus, M. L. and Tanis, C. (2000). The Enterprise System Experience — From Adoption to Success. In R. W. Zmud and M. F. Price (Eds.) FRAMING THE DOMAINS OF IT MANAGEMENT: Projecting the Future Through the Past. Pinnaflex Educational Resources.

McGann, S. T. and Lyytinen, K. (2008). The Improvisation Effect: A Case Study of User Improvisation and Its Effects on Information System Evolution. In the Proceedings of ICIS 2008, Paper 209, 1-14.

Milis, K. and Mercken, R. (2004). The use of the balanced scorecard for the evaluation of Information and Communication Technology projects. International Journal of Project Management 22 (2), 87–97.

MOT Collins English Dictionary (2006). MOT 8.1 Professional (netmot.exe 2.8.4), HarperCollins Publishers.

Mukherji, N., Rajagopalan, B. and Tanniru, M. (2006). A decision support model for optimal timing of investments in information technology upgrades. Decision Support Systems, 42 (3), 1684-1696.

Myers, B. A. (1998). A Brief History of Human-Computer Interaction Technology. Interactions, 5 (2), 44-54. http://doi.acm.org/10.1145/274430.274436 (retrieved 3.11.2010)

Ng, C. S. P. (2001). A Framework for Enterprise Resource Planning Maintenance and Upgrade Decisions. In the Proceedings of the 7th American Conference on Information Systems (AMCIS).

Olson, D. L. (2009) Evolution of and Research in Enterprise Information Systems. Global Journal of Enterprise Information System, 1 (1), 10-20.

Olson, D.L. and Zhao, F. (2007). CIOs' perspectives of critical success factors in ERP upgrade projects. Enterprise Information Systems, 1 (1), 129–138.

Orlikowski, W. J. and Baroudi, J. J. (1991). Studying Information Technology in Organizations: Research Approaches and Assumptions. Information Systems Research, 2 (1), 1-28.

Orlikowski, W. J. and Iacono, S. C. (2001). Research Commentary: Desperately Seeking the "IT" in IT Research – A call to Theorizing the IT Artefact. Information Systems Research 12 (2), 121-134.

Parker, M. M., Benson, R. J. and Trainor, H. E. (1988). Information Economics – Linking *Business Performance to Information Technology*. Singapore: Prentice-Hall.

Peppard, J., Ward, J. and Daniel, E. (2007). Managing the Realization of Business Benefits from IT Investments. MIS Quarterly Executive, 6 (1), 1-11.

Radas, S. and Shugan, S. M. (1998). Seasonal Marketing and Timing New Product Introductions. Journal of Marketing Research, 35 (3), 296-315.

Rashid, M. A., Hossain, L. and Patrick, J. D. (2002). The Evolution of ERP Systems: A Historical Perspective. In L. Hossain, Patrick, D. J. and M. A. Rashid (Eds.) Enterprise Resource Planning: Global Opportunities and Challenges (pp. 1-16). Hershey: Idea Goup Publishing.

Remenyi, D., Money, A. and Sherwood-Smith, M. (2000). The effective measurement and management of IT costs and benefits (2nd edition). UK: Butterworth-Heinemann.

Rose, J. M., Rose, A. M. and Norman, C. S. (2004). The Evaluation of Risky Information Technology Investment Decisions, Journal of Information Systems 18 (1), 53-66.

Ross, J. W. and Beath, C. M. (2002). Beyond the Business Case: New Approaches to IT Investment. MIT Sloan Management Review, 43 (2), 51-59.

Ross, S.A., Westerfield, R.W. and Jordan, B.D. (1998). Fundamentals of Corporate Finance (4th edition). USA: McGraw-Hill.

Rudestam, K. E. and Newton, R. R. (1992). Surviving your dissertation: A comprehensive guide to content and process. California: Sage.

Sawyer, S. (2001). A Market-based Perspective on Information Systems Development. Communications of the ACM, 44 (11), 97-102.

Schwartz, E. S. and Zozoya-Gorostiza, C. (2000). Valuation of Information Technology Investments as Real Options, In the 4th Annual Real Options Conference. London Business School, Real Options Group. http://www.realoptions.org/abstracts/abstracts00.html (retrieved 23.8.2004)

Seacord, R. C., Plakosh, D. and Lewis, G. A. (2003). Modernizing Legacy Systems – Software Technologies, Engineering Processes, and Business Practices. Boston: Addison-Wesley.

Sen, Z. and Y. Zheng (2007). The Relation of CMM and Software Lifecycle Model. In W. Feng and F. Gao (Eds.) the proceedings of the 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2007. Qingdao, China: IEEE Computer Society.

Stake, R.E. (2000). Case studies. In N.K. Denzin and Y.S. Lincoln (Eds.) Handbook of qualitative research (2nd edition). Thousand Oaks: Sage Publications.

Straub, D, Boudreau, M. and Gefen, D. (2004). Validation Guidelines for IS Positivist Research, working paper. Communications of the Association for Information Systems, 13 (24), 380-427.

Straub, D. (2006) The value of Scientometric Studies: An Introduction to a Debate on IS as a Reference Discipline. Journal of the Association for Information Systems, 7 (5), 241-246.

Strong, D. and Volkoff, O. (2010) Understanding Organization-Enterprise System Fit: A Path to Theorizing the Information Technology Artefact. MIS Quarterly, 34 (4), 731-756.

Swanson, E.B. (1976). The dimensions of maintenance. In the International Conference on Software Engineering, San Francisco (pp. 492–495). Long Beach, CA: IEEE Computer Society.

Tamai, T. and Torimitsu, Y. (1992) Software lifetime and its evolution process over generations. In the Proceerdings of Conference on Software Maintenance (pp.63-69).

Tan, Y. and Mookerjee, V.S. (2005). Comparing Uniform and Flexible Policies for Software Maintenance and Replacement. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 31 (3), 238-255.

Tannert, C., Elvers, H-D and Jandrig, B. (2007) The ethics of uncertainty. In the light of possible dangers, research becomes a moral duty. EMBO Reports, 8 (10), 892–896.
http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2002561/
(retrieved 8.3.2011)

Traut, E.M. (2001). The Choice of Qualitative Methods in IS Research. In E.M. Traut (Ed.) Qualitative Research in IS: Issues and Trends. Hershey: Idea Group Publishing.

Turban, E. and Volonino, L. (2010). Information technology for management: transforming organizations in the digital economy (7th edition). Wiley.

Varghese, J. (2003). ROI is not a formula, it is a responsibility. Journal of business strategy, 24 (3), 21-23.

Ven van de, A.H. and Poole. M.S. (1995) Explaining Development and Change in Organizations. The Academy of Management Review, 20 (3), 510-540. http://www.jstor.org/stable/258786 (retrieved 08.05.2009)

Verhoef, C. (2002). Quantitative IT Portfolio Management. Science of Computer Programming, 45 (1), 1-96.

Verville, J., Bernadas, C., and Halingten, A. (2005). So you're thinking of buying an ERP? ten critical factors for successful acquisitions. Journal of Enterprise Information Management, 18(6), 665-677.

Verville, J., Palanisamy, R., Bernadas C. and Halingten, A. (2007) ERP Acquisition Planning: A Critical Dimension for Making the Right Choice. Long Range Planning, 40 (1), 45-63.

Vitharana, P., Bhaskaran, K., Jain, H., Wang, H. and Zhao, L. (2007). Service-oriented enterprises and architectures: state of the art and research opportunities. In the proceedings of AMCIS 2007 (paper 343). Keystone, Colorado. http://aisel.aisnet.org/amcis2007/343 (retrieved 5.12.2011)

Walsham, G. (1995). Interpretive Case Studies in IS Research: Nature and Method. European Journal of Information Systems 4 (2), 74-81.

Walsham, G. (2006). Doing Interpretive Research. European Journal of Information Systems 15 (3), 320-330.

Ward, J., Daniel, E. and Peppard, J. (2008). Building Better Business Cases for IT Investments. MIS Quarterly Executive, 7 (1), 1-15.

Weiderman, N.H., Bergey, J.K, Smith, D.B. and Tilley, S.R. (1997) Approaches to Legacy System Evolution. TECHNICAL REPORT, CMU/SEI-97-TR-014, ESC-TR-97-014. ftp://ftp.sei.cmu.edu/public/documents/97.reports/pdf/97tr014.pdf

Wu, Y. and Saunders, C. (2003). The IS Core – VI: Further along the road to the IT artefact. Communications of the Associacion for Information Systems, 12 (36), 562-567.

Yin, R.K. (2003). Case study research: Design and methods. Thousand Oaks: Sage.

ORIGINAL PAPERS

I

THE LIFE OF INFORMATION SYSTEMS

by

Irja Kankaanpää & Samuli Pekkola, 2011

Submitted to Journal of Information Technology Theory and Application

JOURNAL OF INFORMATION TECHNOLOGY THEORY AND APPLICATION

JITTA

JOURNAL OF INFORMATION TECHNOLOGY
THEORY AND APPLICATION

A Publication of the Association for Information Systems

The Life of Information Systems

First Author
First author's Department
First author's University
First author's e-mail address

Second Author
Second author's department
Second author's university
Second author's e-mail address

Third Author
Third author's department
Third author's university
Third author's e-mail address

etc.

Author affiliation should only be submitted with the final accepted version of the paper.

Abstract:

Lifecycle metaphor is commonly used in defining the life stages of an entity or artefact even though its use has drifted apart from the original meaning. In this paper, the original, biology-based lifecycle definition and the characteristics of life are examined in information systems (IS) context. Particularly the analysis focuses on in-house developed IS and commercial off-the-shelf software (COTS) and on their differences in terms of lifecycle and life characteristics. The results of this conceptual analysis show that as a socio-technical work system, an IS is alive. Hence, the lifecycle and the characteristics of life are applicable in IS context equipping IS management and IS research with an alternative viewpoint for understanding the dynamic nature of IS.

Keywords: information systems, lifecycle, change, development, use and maintenance, implementation, initiation, conceptual analysis, life characteristics

Volume 10, Issue 1, pp., April 2009

………….acted as the Senior Editor for this paper.

The Life of Information Systems

INTRODUCTION

Rapid evolution of technology has implications on the organisation's information systems (IS) portfolio and its management (Benamati and Lederer 2000). Different information systems form a complex, socio-technical aggregate, referred to an information infrastructure (Hanseth and Lyytinen 2007). The management of such large networks of systems poses serious challenges to the IS management that "*are directly or indirectly related to how to manage complex and heterogeneous infrastructures*" (Hanseth and Lyytinen 2004).

Even though Hanseth and Lyytinen (2004) argue that the lifecycle approach is not sufficient for designing complex information infrastructures, we argue that lifecycle approach would be suitable for understanding the mechanical repetition of events within information systems. By following Descartes' reductionist principle (Domsky 2009) the understanding of the whole comes from understanding the parts. Complex systems are not complex because of complex rules but because of the large number of constituents (subsystems) that are connected and interact with each other (Mukherjee 2008). Hence, by reducing the focus to the individual IS level, it is possible to obtain understanding of the organisation's IS as a whole. Under these circumstances, the coherent understanding of the lifecycle of an individual IS becomes critical.

Information systems support organisations and their activities. In this paper, we treat IS as a socio-technical compound, i.e. a work system, where the IS is an integral part of its organisational environment, infrastructure and strategies (Alter 2008a). It is thus "*a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific products and/or services for specific internal or external customers*" (Alter 2008a). This work system executes, manages and coordinates information related work (Lyytinen and Newman 2008).

The stages of Alter's work system model (2008); initiation, development, implementation, and operation and maintenance, follow each other continuously and cyclically. This is similar to nature, where, for instance, seasons follow each other so that the "*what is*" also becomes "*what will be*". We argue that knowing the lifecycle of IS thoroughly, i.e. what are the repeating events of an IS and what are its change mechanisms, we have a basis for further attempts to improve or change it.

The lifecycle metaphor has been adapted to various fields. For example marketing has defined a product lifecycle (Polli and Cook 1969, Kotler et al. 1999), organisations and corporations have different stages of life (Miller and Friesen 1984), and the lifecycle of information, i.e. documents and files, has been studied (Chen 2005). In IS context, the lifecycle metaphor has been used either as an analogy for systems development (see e.g. Davis and Olsen 1985, Currie and Galliers 1999, Kazman et al. 2003, Sen and Zheng 2007) or as a model for systems life stages from birth to death (see e.g. Esteves and Pastor 1999). However, when the lifecycle metaphor was adapted by other fields, it lost some of its original, biology-based meaning. This has been recognised by Van de Ven and Poole (1995) who state that

"*scholars often combine elements of these ideal types [of theories] to explain observed processes of change in specific areas or contexts. However, in such cases it is very easy for the conceptual basis of specific theories to become obscure. As Kaplan (1964) warned, borrowing concepts from different theories without understanding the theoretical 'roots' of these concepts can produce confounded explanations*" (Van de Ven and Poole 1995).

Motivated by this quote, in this paper we will dig into the roots of the lifecycle concept and examine its applicability in IS context. The lifecycle metaphor and life characteristics are adapted from biology and used as a means for studying the IS lifecycle and life characteristics. The paper aims at answering to following questions: Firstly, is IS

CONTRIBUTION

This paper introduces the life cycle concept to information systems context. The characteristics of both life and life cycle provides an alternative approach to conceptualize information systems, particularly systems developed in-house and commercial off-the-shelf systems. The approach helps researchers to understand why certain systems fit with their surrounding ecosystem while some became separated over time. Practitioners benefit the paper by gaining conceptual tools for disentangling IS ecosystems as a whole and as a dynamic entity.

alive in the biology-based sense? If it is, how the life characteristics of an IS relate to the life characteristics of a living organism? Secondly, is the concept of lifecycle applicable in IS context? Thirdly, are there any differences between in-house developed IS software and commercial off-the-shelf (COTS) software? In order to answer these questions, a conceptual-theoretical analysis of the life characteristics and the lifecycle of an IS are carried out.
The paper is structured so that each chapter answers different research questions. The paper ends with discussion and concluding sections.

CHARACTERISTICS OF LIFE

In this chapter, we study whether IS is alive as defined in biology, and if it is, how the life characteristics of an IS relate to the life characteristics of a living organism.

In order to answer these questions, IS is firstly compared to the general characteristics that define life in biology. Although biologists have not been able to agree upon a universal definition for life (c.f. Audesirk and Audesirk 1989, p. 1, Campbell and Reese 2002) they share a view that all living organisms
(1) *have complex and organized structures,*
(2) actively maintain their complex structures, i.e. homeostasis,
(3) grow,
(4) require energy and convert it into other forms,
(5) respond to stimuli,
(6) reproduce,
(7) adapt and evolve (Audesirk and Audesirk 1989);
(8) have a form that fits its function, and
(9) inherit information from their parents (Campbell and Reese 2002).

Next these features are described and examined in IS context in order to study the analogy between with the living organisms and IS.

Complexity and organization of structure: "A basic characteristic of life is a high degree of order. […] Biological organization is based on a hierarchy of structural levels, each level building on the levels below it" (Campbell and Reese 2002). All living things are composed of cells, which are the basic components for the organisation of life (Fried and Hademenos 2001). The cell is the smallest *structural* unit in a living organism that can perform independently the basic *functions* of life (Campbell and Reese 2002). It is the basic unit of structure and function of an organism, and performs "all the activities of life" (Campbell and Reese 2002).

In IS context, the counterpart for a cell is a component; a software module or a system component. At its simplest form, the IS component is a composition of a user, information, and a means for processing the information. "From a structural perspective, an information system consists of a collection of people, processes, data, models, technology and partly formalized language, forming a cohesive structure which serves some organizational purpose or function" (Hirschheim et al. 1995). The components also have a hierarchical structure, as they can be part of a larger aggregate IS, consisting of smaller components each "providing services to the systems at higher layers" (Lankhors et al., 2005, p. 231), or they can be categorized to physical components, processing functions and outputs for users (Davis and Olson 1985). Software has also hierarchy. Each "system is composed of interrelated subsystems, each of which is in turn hierarchic in structure, until the lowest level of elementary subsystem is reached" (Jennings 2001). From structural perspective, the complexity of an IS arises from the large number of interacting components, entities or subsystems (Mukherjee 2008), making IS as work systems rigid and complex (Lyytinen and Newman 2008).

Homeostasis. All living organisms have an optimum state of internal balance within which they operate the best. This includes, for instance, a relatively narrow range of internal temperature, pH and sugar level (Fried and Hademenos 2001). An organism maintains the favourable internal condition constant by means of *homeostasis* (Fried and Hademenos 2001). Homeostasis is an inbuilt regulatory system that controls the physiological conditions and keeps them within tolerable limits. Homeostasis ensures, for instance, that the temperature of the organism does not exceed fatal limit, or that thirst or hunger are experienced when the organism needs fluids or nutrients. (Campbell and Reese 2002).

In IS context, homeostatic includes the evaluation of IS performance against its intended use, and its alignment with the organisation's needs. Self-regulatory activities include maintenance and problem solving, quality control, and the use of guidelines and the best practices.

Growth. During their life, all living things grow (Audesirk and Audesirk 1989) and develop (Arms and Camp 1991). The juvenile phase of a unitary organism, such as a cat, a dog or a human, is typically characterised by growth and development until reaching their maturity. When an animal is born, it does not pose skills, strength and conformation of an adult, but has to develop and grow to become a fully mature individual.

IS exhibit identical growth tendency as any living organism. Individual IS grow and develop locally, with a predictable fashion (Hanseth and Lyytinen 2007). Incremental and iterative development techniques (Larman and Basili 2003) deliver system versions for use at early stage, and, as consequence, the system develops and matures within the user organisation. The stages of growth model describe IS growth as consisting of six stages: initiation, contagion, control, integration, data administration and maturity (Nolan 1979 in Drury 1983). Basically IS grows when new capabilities are integrated into it (Esteves and Pastor 1999).

Energy conversion. All living organisms require nutrients and energy for maintaining their functionality, staying alive and reproduction. An organism exchanges energy with the environment and transforms it from one form to another in order to exist (Campbell and Reese 2002). For example the plans utilize the energy of the sun by photosynthesis and pass that further to animals that eat them (Audesirk and Audesirk 1989 p. 5).
Energy conversion in IS context is related to development efficiency and system effectiveness (Hamilton and Chervany 1981), i.e. how well the given resources (energy) are conversed into benefits throughout the IS lifecycle. Development efficiency refers to the "efficiency with which the IS development (ISD) and operations processes utilize assigned resources (staff, machines, materials, money) to provide the information system to the user" while system effectiveness is the "effectiveness of the users' organisational unit, using the information system in accomplishing their organisational mission" (Hamilton and Chervany 1981). Energy conversion is consequently related to the development, implementation, use and maintenance activities that transform energy for the organisation's benefits.

Responsiveness to stimuli. All living things exhibit responsiveness to both external and internal stimuli. For example the plants bend towards the sunlight (external stimuli) and grow roots downwards (Arms and Camp 1991). Hunger (internal stimuli) drives animals and humans to search for food (Audesirk and Audesirk 1989).
IS responds to its environment and emerging needs by attempting to ensure business-IT alignment (Henderson and Venkatraman 1993). Likewise in nature, in IS context external and internal stimuli stem from the external and internal environments. The external environment includes of legal, political, and cultural considerations (Lyytinen 1987, Lyytinen and Newman 2008). Any request from these sources causes an external stimulus. The internal environment is formed by the immediate organisational environment of the IS including the "goals, tasks structure, volatility, and management style and culture" of the organisation that uses the IS (Lyytinen 1987) or the builds the system (Lyytinen and Newman 2008). Internal stimuli arise from within the IS, from its structure, and its immediate use and development environment. For example a change in the organizational processes, tasks, or technology induces response from the IS. IS management controls how and when the response to different kinds of stimuli takes place.

Reproduction. Living organisms are not immortal thus they need to reproduce to ensure the continuation of life (Audesirk and Audesirk 1989; Arms and Camp 1991). After reaching the reproductive phase an organism gives birth to new organism (Begon et al. 2006).

IS are reproduced in the form of new versions or system upgrades (c.f. Mukherji et al. 2006, Rajlich and Bennett 2000). Reproduction depends on the development efforts of the organization. Technical constructs, such as IT artefacts, are not capable of producing offspring, e.g. new version of an IT application or upgraded IS without human intervention. However, the human part of IS is reproducible.

Inheritance. New organism inherits the traits from its parents. Genetic information that each organism needs for reproduction, survival and development is encoded within its each cell and passed on to the offspring (Arms and Camp 1991). This DNA contains information about the activities of the cells, i.e. information encoding, processing, replication and mutability (Fried and Hademenos 2001).

In IS context, information is inherited at various levels: organisational, technical, data, and structural. At organisational level the experiences from earlier IS implementation activities are inherited in the organisational and ISD practices (Kasvi et al. 2003; Hirscheim et al. 1995; Truex et al. 2000, Päivärinta et al. 2010). Technically, the systems inherit hierarchies and class hierarchies (Crocker and von Mayrhauser 1993) and reusable common features (Coplien et al. 1998). Also data is inherited as legacy systems contain business critical information that has accumulated over the years (Bennett 1995). Basically all deep structures, i.e. fundamental 'choices' of part how its units are organized and their activity patterns and principles, of a parent IS are inherited (Lyytinen and Newman 2008).

Adaptation and evolution: Adaptation and evolution are nature's basic change mechanisms. Adaptation means changes in the organism through which it can better cope with its environment (Audesirk and Audesirk 1989). "Evolution is a genetic change occurring in the population of organisms over many generations" (Audesirk and Audesirk 1989, p. 256). These variations are result of the constant "interplay of environment, variation and selection" (ibid, p. 15). Each new born organism differs slightly from its parents and, if its fit with the environment is better, it is more likely to survive (Darwin 1968). "This preservation of favourable variations and the rejection of injurious variations is called natural selection" (ibid. p. 131).

As organisations change, also IS has to adapt (Armour et al. 1999, Bergey et al. 1997). This takes place either through incremental, gradual adaptation or revolutionary, episodic change (Lyytinen and Newman 2008). The degree of adaptation varies from daily maintenance to large-scale development projects depending on magnitude of the gap. A gap is a state of system that will deteriorate its performance and menace its survival if not taken care for (Lyytinen and Newman 2008).

IS evolution refers to changes in a large IS population, for example moving from in-house developed systems to COTS (Sawyer 2001). This may take place over several generations. Also, legislation can set requirements that trigger system evolution: for example a change in the accounting law sets new rules for the accounting systems. The ones that do not comply with the new regulations die out due to the natural selection of the market forces.

Correlation between form and function. The biological structure "always fits to its function" (Campbell and Reese 2002). For instance a bird's shape is aerodynamic for flying and a rabbit has certain colour for camouflage. This is result of evolution and adaptation to the environment.

The purpose of IS is to support the function and purpose of an organisation (Davis and Olsen 1985; Hirscheim et al. 1995). Therefore, the correlation between form and function is about whether the IS corresponds to the organisational requirements, i.e. does it support business objectives, goals and processes as intended. This also refers to IT-business alignment (Henderson and Venkatraman 1993).

By using above-mentioned characteristics, all things can be classified as living or non-living. As it can be seen, IS fulfills all characteristics of life. They have complex, organized structures, which are actively maintained; they grow, require ener*gy, respond to stimuli, reproduce, adapt and evolve, have a form that fits the function and display heredity. Because the characteristics of life fit with the characteristics of IS, it is be concluded that IS, as a socio-technical system or a work system[1], is alive in the similar sense as organism in nature, and as a corollary, the lifecycle metaphor is applicable in the IS context.

LIFECYCLE

In this chapter, we study whether the lifecycle analogy can be applied in IS context.

A cycle is defined in mathematics as an "orbit that repeats itself every n iterations" (Devaney 2007). In biology, lifecycle is seen as "the generation-to-generation sequence of stages in the reproductive history of an organism" (Campbell and Reese 2002) or as "the series of changes that the members of a species undergo as they pass from the beginning of a given developmental stage to the inception of that same developmental stage in a subsequent generation" (EncyclopædiaBritannica 2007). In other words, lifecycle is an iterative phenomenon of living organisms always ending where it has started. The cycle is complete when an organism reaches the same stage where its ancestor started its cycle, and where the cycle starts over again (e.g. "birth-to-birth").

The lifecycle consists of a sequence of stages that are passed through in a predetermined order (Campbell and Reese 2002). Each stage is characterizes by change or advancement from its preceding stage. A stage cannot be skipped nor returned to. Yet different organisms have different types of stages: for a human they are infant, child, juvenile, adolescent, and adult (Bogin and Smith 1998) and for a bird egg stage, incubation, hatching, chick-feeding period, growth, fledging period, independence, breeding, laying the first egg (Bennett and Owens 2002). Although death is inevitably the last stage in the lifespan of every entity, it is not included in the lifecycle. Death becomes irrelevant after the continuation of the species has been ensured.

Lifecycle theory is one approach to understand the processes of change and development (Van de Ven and Poole 1995). It explains the change as organic growth through specific stages. The lifecycle analogy has been adapted to

[1] It should be noted that if IS is seen purely as a technical construction, the characteristic of reproduction is not completed. However, if Hirschheim et al. (1995) or Alter (2008a) definitions are exploited, IS are indeed alive.

other fields to explain change (Ayres 2004). For instance it has been used to describe the life stages of a product (Polli and Cook, 1969; Klepper 1996; Kotler, Armstrong et al. 1999), organisation (Miller and Friesen, 1984; Van de Ven & Poole 1995; Jawahar and McLaughlin 2001), industrial ecology systems (Ayres 2004), software development (Kazman, Nord et al. 2003) and information (Chen 2005). Interestingly, all these studies regard the concept lifecycle equal to the concept of lifespan, which is a linear period between birth and death (see e.g. Begon et al. 2006). Consequently, instead of describing a lifecycle, these adaptations describe a life history of a phenomenon. For example, product lifecycle has five stages: product development, introduction, growth, maturity, and decline (Kotler, et al. 1999). The biological foundations of the lifecycle theory seem to have disappeared, as discussed by Van de Ven and Poole (1995). Yet the difference between lifecycle and lifespan is significant in IS context. The systems seldom die but evolve and serve as foundations for new system generations (Wieringa et al. 2009; see also Markus et al. 2000). We thus utilize the orthodox lifecycle definition.

Often in the IS context, the e metaphor refers to the stages of ISD lifecycle or systems development lifecycle (SDLC) (e.g. Currie and Galliers 1999; e.g. Kazman et al. 2003; Sen and Zheng 2007). It is common that they do not even encompass the whole life span. For example, SDLC focuses on the development of an IT artefact and leaves later stages, such as the use and maintenance, with less priority (Alter 2008). In order to understand the whole IS lifecycle, this kind of confined approach is not adequate, as it does not capture the IS lifecycle from the organizational perspective (Sawyer 2001).

The comparison between the life characteristics and stages of IS and living organism is presented in Table 1. There the correlation between the life stages of an IS work system and the organisms is evident. Initiation corresponds to conception and development to pregnancy. It can thus be said that the life of an IS begins when organisationally implemented. The use and maintenance stages correspond to the actual life of an IS. Finally, the life of an individual IS ends when it is replaced or abandoned (terminated). The stages between initiation and termination form the lifespan of an IS, even though its features are inherited by the next generation.

Table 1. The analogy between living organism and IS life stages

Information system life stages	**Living organism life stages**
Initiation	Conception
Development	Pregnancy
Implementation	Birth
Use and maintenance	Life (infant, child, juvenile, adolescent, adult)
Replacement, termination	Death

Alter's (2008) work system lifecycle (WSLC) model expands conventional life stage models by defining the IS lifecycle. It treat IS as a socio-technical compound, i.e. a work system, where the IS is an integral part of its organisational environment, infrastructure and strategies (Alter 2008). IS lifecycle is seen as an iterative and continuously evolving socio-technical system with four distinct stages: operation and maintenance, initiation, development, and implementation (Figure 1).

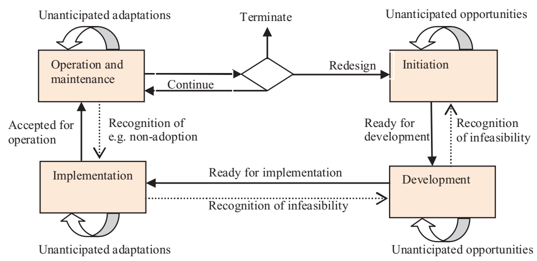

Figure 1. The work system lifecycle model. Adapted from Alter (2008).

After the implementation stage, the process returns to Operation and maintenance phase. A new cycle begins, although possible termination may end the life of the works system. In this perspective, WSLC is a change model where every cycle results a renewed construction of the work system. Yet the distinctive element is continuation: the system continually evolves through iterative cycles. Even though a part of the work system can be 'created from the scratch', the whole work system inherits the characteristics from previous generations.

The change can be viewed as a punctuated socio-technical change which causes a shift from one socio-technical state to another (Lyytinen and Newman 2008). There both social and technical factors influence and are influenced by the change, changing the deep structures of the socio-technical construct. The WSLC model allows both incremental, gradual adaptation and revolutionary, episodic change. However, despite of the type of change, the lifecycle still persists. The lifecycle approach thus essentially depicts change within IS. This change is similarly to nature, where constant growth and development characterize life. Under these premises, the lifecycle metaphor provides a basis for analysing and understanding IS.

IN-HOUSE DEVELOPMENT AND COMMERCIAL SOFTWARE

In this chapter, we study whether and how the lifecycles of two fundamental archetypes of IS, in-house developed IS (in-house IS) and commercial off-the-shelf software products (COTS) (c.f. Markus and Tanis 2000), differ.

In-house IS are developed or tailored for an organisation, either in-house or by a consultant, according to their needs. COTS, on the contrary, are commercial IT applications that are designed to support generic processes. These two types of IS are different in many perspectives. For example, their development goals differ in relation to intended use (c.f. Sawyer 2001): in-house IS are typically developed for a particular purpose (Sabwerhal and Robey 1993) thus supporting predefined and relatively stable functions that are not expected to change much over time. Under the circumstances, their development aims at low maintenance and long lifespan (Truex et al. 1999). Meanwhile COTS are designed either for generic business processes (Markus and Tanis 2000, Sawyer 2001) or for a purpose on smaller market segment, e.g. collaborative IT products (Karsten 1999).

Comparison of life characteristics

Complexity and structure of organisation. As with any software, both in-house IS and COTS have their own structures of organisation. Typically the structure follows composite hierarchy through sub-systems and aggregate systems (Wieringa et al., 2009; Jennings 2001), making it easier to identify different service hierarchies such as physical infrastructure, software infrastructure, business systems, business processes (Wieringa et al., 2009).

Homeostasis. Homeostatic functions are vital for IS survival. For both types of IS, proper IS management, continuous evaluation (Ward et al. 1996), maintenance, and customer and product support are the means of self-regulation (cf. Messerschmitt and Szyperski 2003, p. 332). One symptom of malfunctioning homeostasis is software architecture degeneration (Rajlich and Bennett 2000). It is often related to problems in heritance of critical information related to the source code and architectural decisions (Rajlich and Bennett 2000). Once a system starts degenerating, its maintenance becomes difficult and expensive. Consequently proactive IS management, regular version changes, and continuous evaluation are of utmost importance. The homeostatic functions include evaluation, system support, user support and maintenance.

Growth. The growth of in-house IS is organic, i.e. new features or functionalities are built when needed. Technological or architectural upgrades basically just add new architectural layers, enabling the original system remain operational (cf. Sneed 2006, Bieberstein et al. 2005). The growth of COTS, meanwhile, is dependent on the availability of new modules on the markets. Its growth is consequently modular: new modules, features or capabilities are added into the existing system based on their availability and suitability for the organization (Mukherji et al. 2006). Both types of IS also grow when the number of users increases (volume) and when they are integrated into other systems (complexity), and develop and mature within the organization (see e.g. Cooper and Zmud 1990).

Energy conversion. In-house IS consumes its energy through resources. Those include, for instance, human, technological and financial resources. Developers' and users' development activities are converted into a functioning IS. Through the IS implementation and use, consumed resources are released to the benefit of the organisation. The energy economy forms a closed loop – energy is taken from an organisation and then fed back to it in a different form.

With COTS, the energy flows are different. Energy conversion takes place in the producer's loop where it manifests through the development and release efficiencies, and by developing appropriate software for the customers. Another energy loop takes place in the customer organisation. There the energy is used for selecting a suitable upgrade version, and then purchasing and implementing it. The energy conversion appears as efficiencies in implementation and integration, and by system effectiveness.

Response to stimuli. In-house IS's responsiveness to internal and external stimuli depends on the organizational culture and IS management. In-house systems may respond rapidly to internal stimulus, particularly if it is related to the competitiveness of the organisation. However, *ad hoc* responses easily lead to uncoordinated system changes and chaotic maintenance. COTS, on the other hand, respond slowly, if at all, to internal stimuli due to their market-oriented nature. Collective efforts from the user organisations may be needed to get change requests fulfilled (Markus and Tanis 2000). On the contrary, the responses to the external stimuli, such as changes in legislation, are responded to more actively because they provide business opportunities for new version sales.

Reproduction. Both types of systems are reproduced by developing a new version of the system that replaces its predecessor.

Adaptation and evolution. In-house IS adapts to the changes in its environments, and evolves accordingly. This kind of evolution with large scale changes occurs through development projects. COTS adapt and evolve through versions origin from customer demand, competitive pressure or legislative actions (Rajlich and Bennett 2000). Typically new versions become available faster than the users can adopt them (Sawyer 2001). This indicates that the frequency of the lifecycle of in-house IS is lower to the frequency of COTS since the user organisations have the option of *not* to respond to new versions. COTS thus "live faster" and more intently than the ones that have been developed and nurtured in-house. This may create conflicts if, for example, the vendor recommends frequent upgrades, each requiring changes in the integrated legacy systems.

Correlation between form and function. In-house IS is designed for an organisation. This means that their form is initially made for its function. COTS, on the contrary, are targeted for general fit and for general purpose, not for a particular organisation (Sawyer 2001). The correlation between the form and functions is thus weaker. Yet the adoption of COTS or their upgrades always require changes in an organisation – even though configuration can be used to improve the fit only to some extent (Markus and Tanis 2000).
Inheritance. They types of IS exhibit the inheritance of knowledge, shared data, such as accumulated business data (Bennett 1995) and deep structures (Lyytinen and Newman 2008). In-house IS ideally inherits also the experiences from previous development projects (Avison and Fizgerald 2008, Avison and Fizgerald 2003). Yet in reality, the lessons learned from previous development projects are seldom utilized (Lyytinen and Robey 1999, Avison and Fizgerald 2008). This poses a threat to the continuity and survival of the IS. COTS inherit different experiences from implementation, such as configuration and modification (Chang et al. 2008).
These differences of in-house developed IS and COTS are summarized below, in Table 2.

Table 2. Life characteristics of in-house developed IS and COTS

Life characteristic	***In-house information system***	***COTS***
Structure of organisation	Composite hierarchy: aggregate IS, component IS (Wieringa et al., 2009)	Modular hierarchy: systems, subsystems, subsystem components (Jennings 2001)
Homeostasis	Evaluation, system support, user support and maintenance , IT management	Version upgrades, performance evaluation, IT management
Growth	Functional growth: organic	Functional growth: modular
Energy conversion	Effectiveness of use, development efficiency	Effectiveness of use, efficiency of version upgrades
Response to stimuli	Internal: Depends on the IS management culture of the organisation; potentially efficient and rapid External: Depends on the imperative change requests, i.e. change in legislation	Internal: Depends on the vendor, collective efforts form user organizations may be needed (Markus and Tanis 2000) External: Motivated by external changes that influence the sales opportunities
Reproduction	Development of an improved version	Purchase of a new, suitable version
Adaptation and evolution	Adapts to comply with the needs of the organisation through maintenance. Evolves with an organisation.	Adapts to general market needs through architectural, functional and service arrangements upgrades (Markus and Tanis 2000). Evolves rapidly according to the market demand.
Correlation between form and function	Tight correlation between the functionality and organisational needs	General fit for general processes (Sawyer 2001); configuration is used to improve the fit between COTS and an organisation

		(Markus and Tanis 2000).
Inheritance	Experiences from previous development (Avison and Fizgerald 2008) and implementation projects (Ward, Taylor et al. 1996)	Experiences from implementation, configuration and modification (Chang, Yen et al. 2008)

Comparison of lifecycles

Next the lifecycles of in-house IS and COTS are analysed through the work system lifecycle.

Firstly, the development process and its underlying goals are different. In-house systems are designed to support just the organization that uses it while COTS are targeted for larger audience. The requirements for in-house IS are thus derived from an analysis of the organisation's processes, existing systems, and information needs in co-operation with the developers and future users. COTS on the contrary, are based on market research, product reviews, help-desk call-logs, and user groups (Sawyer 2001). According to Sawyer (2001), their lifecycle is divided into consumer development lifecycle and producer development lifecycle. There the vendor produces "building blocks for IS" and the customer assembles them. The customer is thus dependent on the vendor and their lifecycle, while the vendor is independent on the customer's IS and its lifecycle.

Secondly, the number of stakeholders involved throughout the life of COTS is significantly larger than in the lifecycle of an in-house IS. Yet the relationship between the developers and the users is different in both cases. The development of an in-house IS is a joint effort of developers and users within an organization or a co-operation project with the (internal or external) developer. The development of COTS and consequential IS is a distributed process between distinct consumers, producers and external consultants (Sawyer 2001).

Thirdly, the implementation phases are different. The organisational implementation of COTS is typically more labour intensive than the implementation of an in-house IS. This is because in-house IS can be already be implemented into organization during its development (c.f. Oinas-Kukkonen et al. 2010). In general, the implementation of in-house IS does not require business process changes because the system is already tailored accordingly (Sawyer 2001) – unless it is a part of a planned organisational change. COTS, however, almost always require organisational changes because of its support for general processes. All kinds of changes need to be made in order to comply with new functionality or to accommodate with missing functionality (Sawyer 2001). The customer has also a wide range of options for how to organise the assembly of the COTS: it can be done in-house, with selective external assistance or it can be outsourced[2] (Markus and Tanis 2000, Sawyer 2001). COTS implementation requires different skills and knowledge than the implementation of in-house system (Markus and Tanis 2000). The organisation may easily overlook the analysis of the organisation's processes and requirements as it is focused on "fitting the organisation in the system" (Markus and Tanis 2000; Sawyer 2001).

Fourthly, the goals of ensuring low maintenance and long lifespan (Truex et al. 1999) result that in-house IS changes only when there is a need for change. COTS evolve significantly faster as new versions are produced because of market demand. The evolution of a COTS means that the system undergoes a sequence of iterative

[2] This also applies with purchasing software as service.

changes which result a new product version to the markets (Rajlich and Bennett 2000). The customer has little control on this evolution as the vendor initiates and executes the system changes (Markus and Tanis 2000).

Fifthly, the initiation triggers are different. General speaking both in-house IS and COTS are changed because of three sources: business, technology and vendor (Kankaanpää 2011). However, since the role of system vendor is emphasised with COTS, the vendor (or a company that provides maintenance services on behalf of the vendor) has also more significant role in the change initiation compared to in-house systems (see e.g. Markus and Tanis 2000).

These differences are summarized in Table 3 below.

Table 3. Lifecycle differences between in-house IS and COTS

Lifecycle phase	***in-house IS***	***COTS***
Use and maintenance	Low maintenance (Truex et al. 1999) Use is one of the measures of the IS success (DeLone and Mclean 1992)	Maintenance often requires external consultants. Level of use remains a mystery (Markus and Tanis 2000).
Initiation	Business, technology or vendor driven (see e.g. Kankaanpää 2011)	The driving force of system vendor emphasised (Markus and Tanis 2000). Involvement of vendors, consultants, company executives, and IT specialists (Markus and Tanis 2000)
Development	Requirements specification based on organisation's needs (Sawyer 2001) Unity of lifecycle (Sawyer 2001) User participation during development emphasised (Lynch and Gregor 2004) Development and evolution freedom	Requirements specification based on market demand (Sawyer 2001) Bivalent lifecycle (Sawyer 2001) User organisation dependent on the system vendor (Markus and Tanis 2000) Limited possibilities to influence on the development of new versions
Implementation	Low labour-intensity Business or organisation driven changes	High labour intensity: adaptation to new system, various implementation options Changes in work practices and business processes due to adaptation to the software (Sawyer 2001) User participation during implementation emphasised (Wagner and Newell 2007) Various stakeholders involved, e.g. external consultants

DISCUSSION

In this paper, the life of information systems is studied. Particularly we have examined whether IS are similarly alive as living organisms, and as they are, whether the concept of lifecycle can be used in IS context? We have focused on two archetypes of IS, in-house developed systems and commercial off-the-shelf (COTS) software.

Following Descartes' reductionist principles, understanding the whole comes from understanding the parts. As organisational IS are complex and large and encompass multiples sub-systems, they form a network of interrelated systems, an IS ecosystem. Such complex systems are challenging to depict, analyse and manage. The reductionist principle, however, gives a tool to deduct apparently complex structures into simpler structures. The IS lifecycle approach provides a simple yet comprehensive model for understanding the repeating events that characterise IS life. As the seasons always change in the repeated order, similarly do information systems. This cycle provides a set of attributes that can be used in analysing firstly the components of an IS ecosystem, and secondly the ecosystem as a whole.

The characteristics of life embody critical issues concerning IS management. The shift from in-house developed IS to COTS has brought along collateral changes in the organisations. For example the pace of change has increased, putting demands on updating the IS ecosystem. The organisation itself is no longer in control of the system, particularly COTS. Instead they are dependent on the vendor for changes and modifications. A network of specialists is thus needed to implement and maintain individual components and systems, parts of the ecosystem. Each specialist has specific areas of expertise within the ecosystem, decentralizing the tasks, knowledge, and control. Consequently IS management in the customer organization need to actively monitor, follow, and control not only the IS ecosystem as a whole but also its components and their relationships. This mundane activity of IS management becomes especially challenging when each system, no matter whether in-house developed or COTS, with their own lifecycles, different durations, paces, and stages, are considered. In this situation, the lifecycle approach helps to understand the life of individual components and their relationships.

The *lifecycle* is defined as the sequence of stages an organism undergoes from the beginning of a given developmental stage to the inception of that same developmental stage in a subsequent generation. The basic sequence of stages exists within the lifecycle of an IS: initiation, development, implementation, and use and maintenance stages follow each other in iterative manner (see Alter 2008). The comparison between the lifecycle of a living organism and an information system yields a set of analogies. The decision concerning the initiation of a development (or re-development) project corresponds to conception in living organisms. This is followed by pregnancy where a new living organism is formed or a new IS is developed. As new life is introduced to the world by birth, similarly a new IS is introduced to an organisation by implementation. Life comprises the post-birth developmental stages of an organism, i.e. development from infant stage to child, juvenile, adolescent and adult. In IS context, the actual life stages refer to the use and maintenance. The next cycle of the life of IS begins when the IS life reaches the initiation stage, triggered by need to change.

Both in-house developed IS and COTSs have their explicit lifecycles. However, the lifecycle of an in-house IS is bound to one organisation while the lifecycle of a COTS is distributed over time and stakeholders. When the roles and responsibilities of IS stakeholders are examined it is obvious that the distribution of tasks within the IS lifecycles are different. The development process of an in-house IS encompasses a set of stakeholders that jointly design and construct the system while COTS development process has diverged from the end users and organisation. Because of this fundamental division for "developers" and "consumers", the lifecycle of an in-house developed IS *appears* different than the lifecycle of COTS to the user organisation. The user organisation of an in-house developed IS has at least opportunities to participate in the life of an IS throughout its lifecycle. The sequence of the stages is visible for everyone: use and maintenance, change initiation, development and implementation follow each other logically. Consequently, the user organisation controls the whole process. This is different with COTS. There the distribution of the tasks causes a divided, distributed, and incoherent lifecycle. From the user organisation's perspective, the lifecycle of an ERP system, for instance, is a sequence of implementations as the development activities cannot be seen or influenced (Chang et al. 2008).

If it is assumed that IS are alive, the life characteristics provide fruitful qualities to analyse the living organism of IS. In the following, the attributes common to all life: hierarchical structure, homeostasis, growth, need for energy and energy conversion, response to stimuli, reproduction, adaptation and evolution, correspondence between form and function, and heredity (Audesirk and Audesirk 1989; Campbell and Reese 2002) are discussed within the IS context in order to show the analogy between with living organisms and information systems, and the corollaries of such comparison.

Hierarchical structure. In nature, the smallest unit of life performing the activities of life is a cell. Corresponding unit that can perform basic IS functionalities can be defined as a composition of three parts: information, user, and a data processing mean. This means that an IS cannot exist without external actors or inputs. Its existence can be compared to a symbiotic existence in nature. A data processing mean on its own is merely a vehicle that comes into life when connected to a network of humans and information. In nature, life is highly organised: a group of cells form tissue, various tissue types form organs, and organs form the structure of an organism (Audesirk and Audesirk 1989, p, 3). From IS management point of view, the unit of construction does not have a direct implication. But since all three parts are necessary for the operation of a smallest functional unit of an IS, they appear also in larger scale; on IS level and on IS ecosystem level. An IS ecosystem is a multi-layer hierarchy which defines the relationships between organisation's information systems, how they are integrated and who uses them. IS hierarchy is influenced by organisational hierarchy that defines the relationships between departments, units, and staff members. Software hierarchy defines the relationships between subsystems and their components (Jennings 2001). A well organised hierarchical structure enables faster evolution of a system than non-hierarchical structure (Jennings 2001). To seize this, several tools and approaches, such as enterprise architectures have been proposed to describe, understand and develop hierarchical IS structures.

Homeostasis. Homeostatic functions are vital for IS survival. One symptom of malfunctioning homeostasis is system degradation (Rajlich and Bennett 2000). System degradation origins from critical information related to the source code and architectural decisions (Rajlich and Bennett 2000). Once a system starts degenerating, its mending is difficult. This emphasises proactive IS management and continuous evaluation, maintenance, and customer and product support. In a mixed-IS-ecosystem homeostatic balance is easily disturbed if one system is left uncared for.

Growth. All living things grow and develop. IS grows when new capabilities are integrated into it (Esteves and Pastor 1999). Incremental and iterative development techniques and supplemented system pieces develop and mature the IS within the user-organisation. For example, with ERP systems, the importance of growth is emphasised as often organisations choose to implement only certain modules or functionalities of the whole software (Markus and Tanis 2000). Missing pieces are built and added later. Both in-house and COTS also grow by the use volume, i.e. when the amount of users increases, and by the complexity, i.e. when they are integrated into other systems. Controlling growth becomes a critical issue when the complexity, the risks related to systems integrations, and user resistance increase (Markus and Tanis 2000, Lina and Shaoa 2000).

Need for energy and energy conversion. The energy conversion can be conceptualized through the consumed resources. The developers' and users' efforts are converted into a functioning IS. Through the IS implementation and use, the resources that have been devoted throughout the development phase are released for the benefit of the organisation. This energy economy forms a closed loop – energy is absorbed from an organisation and then fed back to it in a different form. The energy flows of COTS are more complex as the energy conversion takes place in two different loops. In the producer's loop it is manifested through the development and release efficiency, and through the development of appropriate software products. The second energy loop takes place in the consumer organisation where the energy is consumed for selecting, purchasing and implementing software. The rate of energy conversion appears as implementation or integration efficiency, and as system effectiveness. Both developers and consumers are typically very conscious about the energy attributes, particularly financial attributes.

Responsiveness to stimuli. An IS is alive when it is used. This necessitates responsiveness to internal and external stimuli. External stimulus originates from the external environment and includes change requests based on legal, political or cultural considerations. Internal stimulus originates from the internal environment and includes change requests from people, processes, data, models or technology that form the IS. In order to stay alive, an IS has to be flexible enough to respond to these changes in its environments.

Reproduction. The lifecycle results a new life form, a new organism that did not exist before but which inherits features from its ancestors. The lifecycle of an IS (as a work system) results either a transformed IS or a completely new IS. In this sense the lifecycle analogy complies with IS. In IS work system context, the lifecycle is a mechanism for maturation, adaptation and evolution. An IS matures through each cycle, i.e. its level of compliance with the organisation improves. The IS adapts to small scale changes taking place in the organisation (first order change) during its maintenance phase. When more radical changes are needed (second order change) the IS lifecycle either produces a new, upgraded system, or results in adoption of a new system, replacing the existing IS. In order to stay alive, i.e. being used, an IS has to reproduce.

Adaptation and evolution. The purpose of an IS is to support the organisational functions. Thus, when an organisation evolves, its' IS has to evolve with it. If the organizational IS are build in-house, their changes are fewer and smaller as many external changes can be prepared in advance. The organizations have also more control over the evolution and changes. COTS, on the contrary, are highly dependent on the producer-based changes. This means that also the consumer has to response to the producer's evolution. In-house systems thus provide more control for the organisation over the evolution of the IS. However, no matter which types of IS the organization is using, the system starts hindering the organizational development and evolution if an IS is left uncared, i.e. its maintenance and evolution are neglected (Sawyer 2001). It easily becomes a legacy system. To avoid this situation, flexible systems and flexible systems development techniques (Truex et al. 1999), se well as proactive IS management (Armour et al., 1999) are needed. These methods assure that adaptation and evolution, that are vital characteristics for IS survival, are possible.

Correlation between form and function. Correlation between form and function defines how well an IS corresponds to its purpose, i.e. how well it supports the functions of an organisation. This correlation is also known as business-IT alignment (Luftman 2000). High correlation between IS and organisational functions, in general, is desired for two reasons. Firstly, the introduction of an IS that fits well for its function requires little or no changes from the implementing organisation. In-house IS is designed for an organisation, and their needs and functions. Its form is thus made to correlate with its function. On the other hand, the adaptation of COTS almost always requires changes from the adopting organisation (Markus and Tanis 2000). The magnitude of the changes depends on the adopted system's cover area: a small scale system requires minor modification of working procedures while a corporate level enterprise system may require serious business process re-engineering and "renewal of enterprise IT infrastructure" (Markus and Tanis 2000). Secondly, tight correlation between form and function can be a competitive advantage over those companies with loose-fit COTS (see e.g. Carr 2003). An in-house system supports unique business processes that distinguish the organization from others that have adapted general processes defined by COTS. The correlation between form and function operates as an indicator for timing the change. The need for change increases as the gap between the form and the function enlarges. When the gap reaches a certain threshold, it is time for system upgrade (Mukherji et al. 2006). On the other hand, the greater the correlation between form and function, the less there is the need for changes in the organisation. Yet there are reasons for not to strive for high correlation between form and function. In-house systems are slow to evolve because they are designed to operate in stable environment (Truex et al. 1999). If an organisation is volatile, then in-house systems may turn into a stone ledge that hinders the organisation's development (Truex et al. 1999). This can be avoided by the adaptation of COTS, where the correlation is weaker because of their generic nature.

Inheritance. Inheritance concerns information and knowledge transfer. Those have been identified vital for IS survival throughout different stages of its life, particularly at system development (Joshi et al. 2007) and implementation (Ko et al. 2005) stages. Despite of the importance of the inheritance of IS development and implementation, only "very few organizations actively seek further benefits after implementation, or transfer experience of success or failure in benefits delivery to future projects, i.e. they actually learn very little from post-implementation evaluation" (Ward et al. 1996, Lyytinen and Robey 1999). This is apparently paradoxical, because the continuity of the life of an IS is dependent on the inheritance of knowledge. For example, the further development of COTS "starts dying" after the original development team resigns (Rajlich and Bennett 2000). In-house IS are similar but instead of a single product being degenerated and pulled out of the markets, IS, as an ecosystem, is easily turned into legacy systems haunting the maintainers. Also, systems integration and data transfer are vital for the organisation's operations at the events of the system changes (replacement, renewal, expansion). The implications from failed inheritance are consequently manifold and serious. From the IS management point of view, a modern IS portfolio with a mixture of in-house IS and COTS is a challenging setting. The evolution of in-house IS requires the knowledge of that particular system while the upgrade of COTS necessitates experiences from similar implementation and configuration projects. Having all the knowledge in-house obviously expands the size of IS personnel. Inheritance is thus indirectly connected to the size and skill-profile of the organisation's IS department. The challenge is to have a right combination of staff that can handle both COTS and in-house systems within the budget frames. Similarly systems integration and data flows pose challenges as they are intertwined into an entangled ecosystem through interfaces, dependencies, models, and architectures – all inherited from the previous systems generations.

Based on the discussion above it is clear that the biological life characteristics are determinants of IS life. The significance of the life characteristics is summarised as follows:

- Hierarchical structure: Hierarchic structure forms the basis for the controllability of the complexity. In nature the great complexity of all life is possible due to its hierarchically organised structure. In IS context the same applies. Well organised architecture is the backbone of functional and operational IS. This underlines the importance of understanding and knowing the information systems architecture, preferably even the enterprise architecture.
- Homeostasis: Maintenance and evaluation ensure that an IS is operating within its optimum limits and that the IS does not start to degenerate. IS personnel and resources management is a part of the homeostatic functions thus their knowledge, skills and activities are emphasised.
- Inheritance: In IS context the inheritance takes its shape in the form of kknowledge transfer. It occurs at various levels (organisational, group, individual) including e.g. tacit knowledge and experiences from development and implementation projects for the benefit of future projects, and accumulated data within information systems. Thus, one has to learn from the past decisions and their consequences.
- Correlation between form and function: This refers to the importance of business-IT alignment. It is also an indicator for timing a planned change. The need for change increases as the gap between form and function widens.
- Reproduction: Reproduction is the basis for continuity of IS lifecycle.
- Adaptation and evolution: An IS has to adapt to the changing organisation and external demands. Also a direct managerial implication is to show caution when integrating systems with significantly differing lifecycle speed. Technically, adaptation takes place through system upgrades and integration with existing systems

or of new modules (Chang et al. 2008) in form of regular maintenance. If the maintenance fails, the system will become will easily become a burden to the organization.
- Growth: Growth enables the IS to develop and mature through its life stages.
- Need for energy and energy conversion: Continuous inputs (human resources) are needed in order for the IS to stay operational. From the organization's sake, it is optimal to converse more energy (i.e. add value) than consume it (waste resourses).
- Response to stimuli: In order to stay alive, an IS has to be flexible enough to respond to the changing requirements from its environments, both external and internal.

Based on the discussion above we suggest the following issues to be considered in IS management in the light of lifecycle thinking:
- Control: The control over an in-house IS should be centralised and only a few stakeholder groups should be involved in its maintenance. This limits the scattering of the development, implementation and use activities, and increases the ability to control and understand the whole ecosystem, and consequently decreases risks, unnecessary variations and unexpected surprises.
- Governance: It should be considered how the governance and responsibilities of an IS are distributed, i.e. what is managed and maintained in-house, what is outsourced and what is the responsibility of COTS vendor.
- Path dependence: When planning a new system, the development alternatives (in-house or COTS) are often exclusionary. One has to choose between the alternatives and bear the consequences of the choice. Hence, the existing IS defines what are the future development alternatives (Hanseth and Lyytinen 2007). In an IS ecosystem, in-house and COTS based IS have been integrated within the limits of the previously made choices, the "path".

Many authors (e.g. Markus and Tanis 2000; Sawyer 2001) focus on explaining the differences between COTS and in-house systems. In reality IS ecosystems are seldom such homogenous and, therefore, the analysis of the mixed-IS-ecosystem and its different lifecycles is valuable for both practitioners and researchers. Both types of IS have their managerial requirements on staff skills and knowledge and overall IS governance. One necessitates familiarity with different outsourcing options while the other strong ISD development and maintenance skills. But what happens to the tasks and responsibilities when IS ecosystem consists of mixture of systems? How can one define what skills are needed more than others, and how can the traditional and modern IS management operate jointly? Are they exclusive ways of working or can they be combined? The challenge is to find a balance between the new and old requirements, COTS and in-house systems and to utilize them in the best possible way. It is suggested that our lifecycle approach provides an alternative way of understanding information systems and their change mechanisms. Both in-house IS and COTS have explicit lifecycles that are not identical as commonly used concept; the stages of life, which is often used as a synonym to lifecycle. The essence of lifecycle has not thus been realised in large scale. From the managerial perspective, the lifecycle approach provides understanding and ability to identify the events that are repeated in every IS after IS. This provides opportunities for continuous learning and improvement. Lifecycle is a central concept when managing an individual IS or the IS ecosystem, i.e. a collection of information systems, which all have their individual lifecycles within an organisation.

CONCLUSIONS

In this paper, a lifecycle concept has been presented following its original definition in natural sciences. The structured analysis helps us to understand the "life" of information systems, particularly in-house developed IS and COTS. Such analysis reveals that IS shares surprisingly great amount of the features of living things. Based on these premises, it is theorised that the lifecycle analogy is suitable for IS context. It is especially suitable for understanding the iterative, repetitive events in IS life, namely use and maintenance, initiation, development and implementation.

The lifecycle analogy provides a means for understanding both IS ecosystem as a whole, and of its individual systems. This has also managerial implications for the management of mixed-IS ecosystem. It complements current approaches in managing IS and their relationship with the organisation. While enterprise architecture describes entities that have a lifecycle and that functions in a shared ecosystem, the lifecycle approach provides insights to the life stages and life prerequisites of an individual IS. In this perspective, IS and their aggregates can be compared to ecological structures. Their management would greatly benefit if they are understood as such. Often individual information systems are at the different phases of their lifecycle. One system can be near retirement while another is just introduced in the organisation. Thus, they most likely have different correlation between the form and function, and the interaction mechanisms with the environment differ since the level of adoption is different. The users are familiar with the old system while the newcomer may still be going through a mutual adaptation process. Under the circumstances, the IS management may benefit from an alternative viewpoint when trying to understand the situation and dependencies.

During the last decades, the trend has been that the development of IS has shifted from in-house systems to COTS-based systems. The main difference between in-house development and commercial software development is that in-house development aims at developing a particular IS for a particular purpose while commercial software development aims at developing a software for generic purpose (Sawyer 2001). As simple as it may sound, a software application is not an IS as such. In order to become an IS (or a part of an IS) a software has to implemented in an organization and integrated into its existing systems, infrastructures, and processes. Only after implementation it will come into “life”.

In this paper, we have conducted as a conceptual analysis on the lifecycle concept. It is obviously limited to theoretical investigation. Thus, there is a clear need for further research to empirically test and validate the presented lifecycle approach. During the literature analysis, it was noticed that the concept of lifecycle has been widely adopted in fields of marketing, organization sciences, and IT, and as this happened its meaning had changed from cyclic “birth-to-birth” approach to linear “birth-to-death” approach. It is suggested, that that staying with the original definition is important since it encapsulated the vital element of iteration, the cycle. It is proposed, that one problem related to the difficulty of IS management is that its entities, i.e. business, information, IS, and infrastructure, are perceived from linear life to death perspective. However, data, information, knowledge, and experiences do not disappear once a system reaches the end of its life span. Instead, they are used later, when new systems are introduced. Linear approach thus unnecessarily narrows down the view on the IS and the whole IS ecosystem. When the perspective is switched to lifecycle view, a new dimension, which enables the investigation of the relationships between the entities, is opened. This proposition is so far lacking empirical evidence and hence we recommend it as a future research topic. Other potential topics include the empirical investigation of the evolution and management of mixed-IS ecosystem (i.e. an IS ecosystem that has both in-house and COTS based systems) as a whole, impact analysis (or butterfly effect) study of a single change in an IS ecosystem, and the management of the evolution of an IS ecosystem.

Our primary contribution to IS research is the introduction of the lifecycle approach to the IS context. It provokes questions and inspires further research. Our main contribution to IS practice is the analysis of the managerial implications of the importance of life characteristics and the iterative nature of IS lifecycle.

REFERENCES

Alter, S., “Defining information systems as work systems: implications for the IS field. *European Journal of Information Systems* 17(5), 2008. pp.448-469.

Armour, F. J., Kaisler S. H. and Liu S.Y., "A Big-Picture Look at Enterprise Architectures." *IT Professional* 1(1), 1999, pp. 35 – 42.

Arms K. and Camp P. S., *Biology: A journey Into Life*, Saunders College Publishing, U.S.A., 1991.

Audesirk G. and Audesirk T., *Biology: life on earth*. 2nd edition. Macmillan Publishing Company, U.S.A. 1989.

Avison, D. & Fitzgerald, G., *Information systems development*, 4th edition. McGraw-Hill, UK. 2008.

Avison, D. & G. Fitzgerald. “Where now for development methodologies?” *Communications of the ACM* 46(1). 2003.

Ayres, R. U. "On the life cycle metaphor: where ecology and economics diverge." *Ecological Economics* 48(4), 2004, pp. 425-438.

Begon M., Townsend C. R. and Harper J. L., *Ecology: From Individuals to Ecosystems*. 4th edition, Blackwell Publishing, Singapore, 2006.

Benamati, J. and A. L. Lederer. “The emerging it group and rapid IT change”. *ACM SIGCPR conference on Computer personnel research*, ACM, New York, USA. 2000

Bennett, P. & Owens, I. *Evolutionary Ecology of Birds: life histories, mating systems, and extinction*. Oxford University Press. 2002.

Bennett, K., “Legacy Systems: Coping with Success”. *IEEE Software*, 12 (1), 1995, pp. 19-23.

Bergey JK, Northrop LM and Smith DB, *Enterprise Framework for the Disciplined Evolution of Legacy Systems*. Technical report, CMU/SEI-97. 1997.

Bieberstein N., Bose S., Walker L. and Lynch A. “Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals”. *IBM SYSTEMS JOURNAL*, 44 (4), 2005, pp. 691-708

Bogin, B. and H. Smith “Evolution of the human life cycle”. *American Journal of Human Biology* 8(6), 1998. pp.703-716.

Campbell, N. A. and J. B. Reese, *Biology*, Benjamin Cummings. 2002.

Carr, N.G. “IT doesn't matter”. *Harvard business review*, May, 2003. pp.41-49.

Chang S, Yen CD, Huang S and Hung P, “An ERP System Life Cycle-Wide Management and Support Framework for Small- and Medium-Sized Companies”. *Communications of the Association for Information Systems* 22(15), 2008, pp. 275-294.

Chen, Y. “Information Valuation for Information Lifecycle Management”. *Second International Conference on Autonomic Computing (ICAC)*, IEEE Computer Society. 2005.

Cooper, R.B. and R. W. Zmud. “Information technology implementation research: A technology diffusion approach.” *Management Science*, 36 (2), 1990, pp.17-31.

Coplien, J., D. Hoffman, et al., "Commonality and variability in software engineering." *Software IEEE* 15(6), 1998, pp. 37-45.

Crocker, R. T. and A. von Mayrhauser. “Maintenance support needs for object-oriented software”. *Seventeenth Annual International Computer Software and Applications Conference COMPSAC 93*.1993.

Currie, W. L. and B. Galliers. *Rethinking management information systems*, Oxford. 1999.

Darwin, C. *The Origin of Species by Means of Natural Selection or the preservation of favoured races in the struggle for life*. Middlesex, Great Britain, editted by J. W. Burrow. Penquin Books. 1968.

Davis, G. B. and M. H. Olsen. *Management Information Systems: Conceptual foundations, structure, and development*. Singapore, McGraw-Hill. 1985.

DeLone W.H. and McLean E.R. “Information Systems Success: The Quest for the Dependent Variable.” *Information Systems Research*, 3(1), 1992, pp. 60-95.

Devaney, R. L. "Mathematical Glossary (On-Line)." Retrieved November 30, 2007, from http://math.bu.edu/DYSYS/explorer/def.html. 2007.

Domsky M. “The intelligibility of motion and construction: Descartes’ early mathematics and metaphysics 1619–1637.” *Studies in History and Philosophy of Science*, 40, 2007. pp. 119–130

Drury D. H. “An Empirical Assessment of the Stages of DP Growth”. *MIS Quarterly*, 7(2), 1983, pp. 59-70.

EncyclopædiaBritannica. "Life cycle - Encyclopædia Britannica Online." Retrieved October 23, 2007, from http://www.britannica.com/eb/article-9048202. 2007.

Esteves J.M. and Pastor J.A., “An ERP Life-cycle-based Research Agenda.” *Proceedings of the First International workshop in Enterprise Management and Resource Planning: Methods, Tools and Architectures*, Venice, Italy. 1999, pp. 359-371,

Fried, G. H. and Hademenos, G. J. *Schaum's Easy Outline: Biology*, McGraw-Hill Professional Book Group, ISBN: 9780071369718 (retrieved 27.10.2010) http://site.ebrary.com/lib/jyvaskyla/docDetail.action?docID=5004553 2001

Hamilton, S. and N. L. Chervany. "Evaluating Information System Effectiveness - Part I: Comparing Evaluation Approaches." *MIS Quarterly* 5(3), 1981, pp. 55-69.

Hanseth, O., and Lyytinen, K., “Theorizing about the Design of Information Infrastructures: Design Kernel Theories and Principles,” *Sprouts: Working Papers on Information Environments, Systems and Organizations*, 4 (4), 2004, pp. 207-241. http://sprouts.case.edu/2004/040412.pdf

Henderson, J.C. and Venkatraman, N.. “Strategic Alignment: Leveraging information technology for transforming organizations”, *IBM Systems Journal*, 32(1), 1993, pp. 4–16.

Hirschheim R, Klein H and Lyytinen K., *Information Systems Development and Data Modeling, Conceptual and Philosophical Foundations*. Cambridge University Press, Cambridge. 1995.

Jawahar I.M. and McLaughlin G.L. “Toward a Descriptive Stakeholder Theory: An Organizational Lifecycle Approach”. *The Academy of Management Review*, 26(3), 2001, pp. 397-414.

Jennings, N. R. "An agent-based approach for building complex software systems." *Communications of the ACM* 44(4), 2001, pp. 35-41.

Joshi K.D., Sarker S. and Sarker S. “Knowledge transfer within information systems development teams: Examining the role of knowledge source attributes”. *Decision Support Systems*, 43 (2), 2007, pp. 322-335,

Kankaanpää, I., “Information system’s life cycle: Interpretative study on CIO perceptions”. *Submitted* 07/2011. 2011

Karsten, H. “Collaboration and Collaborative Information Technologies: A review of the evidence”. *The DATA BASE for Advances in Information Systems*, 30(2), 1999, pp. 44-65.

Kasvi J.J.J, Vartiainen M. and Hailikari M. “Managing knowledge and knowledge competences in projects and project organizations”. *International Journal of Project Management* 21, 2003, pp. 571–582.

Kazman, R. R., Nord, R. L. and Klein M.H. *A Life-Cycle View of Architecture Analysis and Design Methods*. Technical Note (CMU/SEI-2003-TN-026). 2003.

Klepper S., “Entry, Exit, Growth, and Innovation over the Product Lifecycle.” *The American Economic Review* 86(3). 1996, pp. 562-583

Ko D-G., Kirch L.J. and King W.R. “Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations”. *MIS Quarterly*, 29 (1), 2005, pp. 59-85.

Kotler, P., G. Armstrong, et al.. *Principles of Marketing*, Prentice Hall Europe. 1999.

Larman C. and Basili R. V. “Iterative and Incremental Development: A Brief History”. *Computer*, IEEE Computer Society, June 2003, pp. 47-56

Lederer, A.L. and Salmela, H., “Toward a theory of strategic information systems planning”, *Journal of Strategic Information Systems* 5(3), 1996, pp. 237–253.

Lina, W. T. and Shaoa, B.B.M. "The relationship between user participation and system success: a simultaneous contingency approach." *Information & Management* 37: 2000 pp. 283-295.

Luftman, J. "Assessing Business Alignment Maturity." *Communications of AIS* 4 (14). 2000

Lynch, T. and Gregor. S. “User Participation in Decision Support Systems Development: Influencing system outcomes”. *European Journal of Information Systems* 13, 2004. pp. 286-301.

Lyytinen K., and Robey D. "Learning Failure in Information System Development", *Information Systems Journal*, 9(2), 1999, pp. 85-101

Lyytinen, K. "Different perspectives on information systems: problems and solutions." *ACM Computer Surveys* 19(1), 1987, pp. 5-46.

Lyytinen, K. and Newman, M. “Explaining information systems change: a punctuated socio-technical change model”. *European Journal of Information Systems* 17(6), 2008, pp. 589-613

Markus ML and Tanis, C., “The Enterprise System Experience — From Adoption to Success”. *Framing the domains of IT management: Projecting the Future Through the Past* (Zmud RW and Price MF, Eds), Cincinnati, Pinnaflex Educational Resources. 2000.

Messerschmitt, D.G. and C. Szyperski. *Software Ecosystem*. The MIT Press. 2003.

Miller, D. and Friesen, P.H. “A longitudinal study of the corporate life cycle”. *Management Science* 30(10), 1984, pp. 1161-1183.

Mukherjee I. “Understanding Information System Failures from the Complexity Perspective”. *Journal of Social Sciences* 4(4), 2008. pp. 308-319.

Mukherji, N., B. Rajagopalan and M. Tanniru. “A decision support model for optimal timing of investments in information technology upgrades”. *Decision Support Systems*, 42, 2006, pp.1684-1696.

Oinas-Kukkonen, H., Hohtari, S. and Pekkola, S. “Organizing End-User Training: A Case Study of an E-Bank and its Elderly Customers”. *Journal of Organizational and End User Computing* 22(4), 2010. pp. 95-112

Päivärinta, T., Sein, M. K. and Peltola, T. “From ideals towards practice: paradigmatic mismatches and drifts in method deployment”. *Information Systems Journal*, 20(5), 2010, pp. 481 – 516.

Polli, R. and V. Cook. "Validity of the Product Life Cycle." *The Journal of Business* 42(4), 1969, pp. 385-400.

Rajlich T.V. and Bennett K.H. ”A Staged Model for the Software Life Cycle”. *Computer* 33(7), 2000, pp. 66-71.

Sabwerhal, R. and D. Robey. "An empirical taxonomy of implementation process based on sequence of events in information system development." *Organization Science* 4(4), 1993, pp. 548-576.

Sawyer, S. “A Market-based Perspective on Information Systems Development”. *Communications of the ACM*, 44(11), 2001, pp. 97-102.

Sen, Z. and Y. Zheng. “The Relation of CMM and Software Lifecycle Model”. *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, IEEE Computer Society. 2007.

Sneed, H.M. “Integrating legacy software into a service oriented architecture”. *The American Economic Review* 86(3), 2006, pp. 562-583

Truex D., Baskerville R and Travis J., “A methodical systems development: the deferred meaning of systems development methods”. *Accounting, Management & Information Technology*. 10, 2000, pp. 53–79.

Truex, D. P., R. Baskerville, et al. “Growing Systems in Emergent Organizations”. *Communications of the ACM*, 42(8), 1999, pp. 117-123.

Van de Ven AH and Poole MS, “Explaining development and change in organisations”. *Academy of Management Review*, 20(3), 1995, pp. 510-540.

Wagner, E.L. and S. Newell. “Exploring the Importance of Participation in the Post-implementation Period of an ES Project: A Neglected Area”. *Journal of the Association for Information Systems*, 8 (10), 2007, pp. 480-479.

Ward, J., P. Taylor, et al. "Evaluation and realisation of IS/IT benefits: an empirical study of current practice." *European Journal of Information Systems* 4(4) 1996.

Wieringa R.J, van Eck P.A.T. and Krukkert D. ”Introduction to Enterprise Architecture”. *Enterprise Architecture at Work - Modelling, Communication, and Analysis*, M. Lankhorst (ed.), Springer-Verlag Berlin Heidelberg, Germany. 2009.

II

INFORMATION SYSTEM'S LIFE CYCLE: INTERPRETATIVE STUDY ON CIO PERCEPTIONS

by

Irja Kankaanpää, 6.7.2011

Submitted to Scandinavian Journal of Information Systems

Information system's life cycle:

Interpretative study on CIO perceptions

Abstract. Life cycle theory explains change and development in organizations. In this study it was utilized as a framework to explain change in information systems (IS). The conduct of the study follows the interpretative tradition. Chief information officers (CIO) of large organizations were interviewed in order to capture their perceptions on IS life cycle. The findings indicate that the IS life cycle is perceived either as a linear sequence of stages, as life time or as a shared life cycle with organization's processes. The interpretation of the findings indicates that these perceptions are related to the perception of the IS itself. The underlying IS perception correlates to technical and socio-technical view. When viewed as a technical system, the IS consists of software and hardware, and its life cycle portrays a sequence of stages from beginning to end with certain duration. When viewed as a socio-technical system, the IS is embedded in the organizational processes and evolves as the organization changes with no beginning or end.

Keywords: Information system, life cycle, evolution, IT artifact, CIO perception

1. Introduction

Life cycle theory is one of the major theories that have been used to explain the development of organizations (Van de Ven and Poole 1995, Miller and Friesen 1984). It approaches change and development from the viewpoint of organic growth where predefined stages follow each other (Van de Ven and Poole 1995). IS life cycle is commonly considered as a sequence of stages from cradle to grave, i.e. from the initial development to retirement (Esteves and Pastor 1999, Avison and Fitzgerald 2003, Chang et al. 2008). In reality, however, the life of an IS is seldom a linear sequence of stages. Instead, it is rich with numerous iterations of upgrades (Beatty and Williams 2006), integrations (Markus and Tanis 2000, Sneed 2005) migration (De Lucia et al. 2008) by which the capabilities and functionality of the system are extended or improved after the initial development (Rajlich and Bennett 2000). Upgrades are needed at regular intervals in order to keep IS aligned with changing organizational requirements (see e.g. Mehra and Seidmann 2006, Lyytinen and Newman 2008).

The definition of IS life cycle is dependent on the definition of IS itself. Similarly, the definition of the IS, in turn, is dependent on the discipline and position of the person (the perceiver) towards it (Orlikowski and Iacono 2001). As an example, management and social science focuses on the effects of IS while in computer science the main focus is in the computational capability of IS (Orlikowski and Iacono 2001). Benbasat and Zmud (2003) argue that one of the major problems with the IS discipline is that it studies issues and phenomena that are remotely related to the IS and revolve around the IS but that *are not* the IS (or the IS based system). Orlikowski and Iacono (2001) argue that in IS research the core of IS, i.e. IT artefact, is taken for granted. As a consequence many critical questions are left unasked. Hence, Orlikowski and Iacono (2001) call for "*IS researchers begin to theorize specifically about IT artifacts*" and "*to take technology as seriously as its effects, context, and capabilities*". In other words, they encourage the search for the missing IT artefact within IS research and setting that as the focus of the IS research. Motivated by the call of Orlikowski and Iacono (2001), the IT artefact is approached here from the perspective of the life cycle theory. The objective of this paper is to study how the IS life cycle is comprehended in organisations by CIOs. In order to answer to this objective, both theoretical and empirical investigation was carried out. Interpretative research method was used in order to analyze the nature of IT artefact behind the IS life cycle definitions found in the empirical study.

The rest of the paper is organized as follows. Firstly, the related literature is presented in section 2. The research method is described in section 3. The empirical findings are reported in section 4 and their interpretation in section 5. Discussion is presented in section 5. Conclusions are made and limitations of the research are discussed in section 6.

2. Related literature

In IS research, the life cycle concept has been used to model IS development (ISD) process (see e.g. Avison and Fitzgerald 2003, Davis et al. 1988). It describes the IS life cycle as a linear sequence of stages consisting of feasibility study, system investigation, systems analysis, system design, implementation, and review and maintenance stages (Avison and Fitzgerald 2003). Examples of ISD life cycle are numerous, the best known being the waterfall model (Davis et al. 1988). These models emphasize the development and use of technology (Orlikowski and Iacono 2001). The life cycle analogy has been used also to describe the acquisition and organizational deployment of commercial systems such as enterprise resource planning (ERP) systems (see e.g. Fenema et al. 2007) or enterprise systems (ES) (see e.g. Nordheim and Päivärinta 2006). An ERP system's life cycle is commonly described as a linear sequence of stages, e.g. adoption decision, acquisition, implementation, use and maintenance, evolution, retirement stages (Esteves and Pastor 1999). The problem with the linear, development or acquisition oriented models is that they lack continuity and the ability to describe IS change and its relation to organizational change (see Lyytinen and Newman 2008, McGann and Lyytinen 2008). They fail to depict IS life cycle and evolution from organizational perspective.

In this paper, the work system life cycle (WSLC) is used as research framework as it binds IS into its organizational context and through this union defines IS life cycle. Here, IS is approached from socio-technical perspective and defined as a work system (Alter 2008a): *"A work system is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific products and/or services for specific internal or external customers. An IS is a work system whose processes and activities are devoted to processing information, that is, capturing, transmitting, storing, retrieving, manipulating, and displaying information. Thus, an IS is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce informational products and/or services for internal or external customers."*

According to this definition, IS life cycle is connected to the processes, procedures and environments in which it is used. IS is viewed as an integral part of a work system and its life cycle is hence depicted through them. Work system life cycle (WSLC) models (Alter 2008b) four stages, i.e. operation and maintenance, initiation, development and implementation that recur in the aforementioned order (Figure 1). In the WSLC the IS evolves through iterations (Alter 2008a). A cycle is preceded by a decision to initiate system redesigning that leads to initiation of a development project. This paper is based on the assumption that the WSLC model depicts the IS life cycle within an organization more accurately than linear life cycle models.

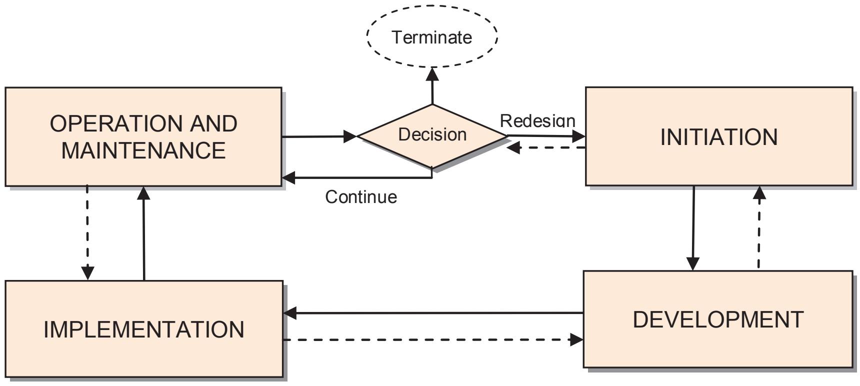

Figure 1. Work system life cycle (Alter 2008b)

Theories have been used to explain the phenomena of IS life cycle and how IS relates to its immediate (and remote) environment. Because IS as a research field is lacking its own theories the phenomena has been explained by borrowed theories and by using analogies from other fields of study, including organizational science, economics, management science (Benbasat and Zmud 2003) organic biology, ecology, and social studies (Porra 1999). An IS has been explained as an organic living entity, a part of an ecological system or a colony (Porra 1999). These theories do not study IS life cycle *per se* but are important to note because they touch the life cycle closely and include it implicitly.

Change is a central element in IS life cycle. Software systems have to evolve, i.e. go through progressive change, in order to remain satisfactory to the community they serve (Lehman & Ramil 2003). Theories that have been utilized to describe the life cycle of an IS include punctuated equilibrium theory, episodic system change theory, systems theory, socio-technical systems theory and process theory (Lyytinen and Newman 2008). Since information systems are closely linked to the organization in which they operate, also theories explaining organizational change are close to explaining IS life cycle. The four major theories explaining change include life cycle theory, evolutionary theory, teleological theory and dialectical theory (Van de Ven and Poole 1995).

The challenge of applying analogies and theories from other disciplines is, however, that when they are separated from the original context they get generalized and corrupted. When, e.g. analogies from biology are attempted to be adapted to non-living (technical) objects, their applicability cannot be taken for granted because the targets of observation are not comparable. An IS is a socio-technical system and, hence, cannot be described solely with technical theories, for example, or with social theories because it is a mixture of both (Porra 1999). Therefore, Porra (1999) suggests that a combination of theories or a new type of theory, such as colonial systems theory, is needed.

The literature analysis reveals that the IS field is lacking a concrete definition for IS life cycle. One reason for this is that the information system life cycle is directly related to the definition of information system. As long as the concept of IS itself remains vague, the definition of IS life cycle is obscure (see the variety on IS definition in Alter 2008a). The problem with defining IS is that it has been used to describe "different types of objects that have many aspects in common" (Alter 2008a). Another reason is that the definition of IS life cycle is dependent on the perspective (e.g. implementation, development) one adopts towards the IS and the system boundaries, i.e. what one includes within the definition of IS (work system view, change theories). The *perspective* in its turn is dependent on the *perceiver* (Orlikowski and Iacono 2001). Consequently, it is necessary to investigate how the phenomenon of IS life cycle is conceptualized in real life by including the perceiver and their perspective towards the IS.

3. Research method

This research was conducted according to the interpretative tradition (Orlikowski and Baroudi 1991). Interpretative research aims at capturing interviewees' perceptions and making interpretations of them (Walsham 1995, Walsham 2006) and through this seeks understanding of the studied phenomenon (Orlikowski and Baroudi 1991). The principles of interpretative research (Walsham 1995, Walsham 2006) were applied in the following way: Firstly, the style of involvement is explained. Secondly, the collection of the field data is reported in detail, including the way of gaining access to the research field, the reasons for the selection of the research sites, the number of interviewees, the organizational position of the interviewees, time period of conducting the research and what other data sources were used. Thirdly, the reporting of data analysis includes description of recording practices of the interviews, description of the data analysis and how the iterative process between the collected data and theory took place. The principles of work in different countries (Walsham 2006) was not applied since the study was conducted within one country.

The style of involvement was the one of an outside researcher without action or involvement in the field. This means that the researcher approached the field through interviews with the benefit of a fresh outlook of the field and with minimum identification with the subjects (Walsham 2006).

The access to the field was gained through persistent contacting of potentially suitable organizations. The target group of the interview was the chief information officers (CIO) of large private enterprises and public organizations. They represent organizations that utilize IS in their operations, i.e. the user organization's perspective. The development perspective was excluded from the empirical study and hence no vendors or system producers were interviewed. CIOs were selected as interview subjects because they have the overall responsibility of the IT management in organizations (Varghese 2003). The selection of large organizations as research targets was based on the assumption that large organizations have potentially more information systems and, as a consequence, larger experience base on the research topic than small and medium sized organizations. The findings from large organizations were anticipated to be useful for the whole community of IS practitioners.

The selection of the organizations was done according to the available resources. As a consequence the search of targets was geographically limited to central and southern Finland, close to the research facilities. However, the amount of included organizations (13) is suitable for qualitative research (Denzin and Lincoln 2000) where the amount of, or the selection of the cases is not as meaningfull as the opportunity to learn from the cases of the studied phenomenon (Stake 2000). The final sample represented national and international companies and organizations from eight different industries. The field of industry was not considered relevant because each of the organizations represents a larger group of IS user-organizations despite of the selection method (see Stake 2000).

The research process started with literature review. This was followed by the formulation of the interview structure and questions. Interview questions were formulated on the basis of a literature survey and a theoretical analysis on IS life cycle. Interview, as the primary data source in interpretative research (Walsham 1995), was considered as the most suitable means for the empirical investigation because the topic was new. Semi-structured interview was used as the data collection method as it gives interviewees the freedom to express their own views while keeping the conversation on the topic (Fontana and Frey 2000). The interview was tested before the actual data collection. Three senior researchers reviewed the questions and gave their suggestions for improvements. The original interview questions were formulated in English but the actual interviews were conduced in Finnish. No other data sources was used in addition to the interviews.

Data acquisition proceeded as follows. An inquiry for participation in interview was sent by email to 37 large organizations that were found through internet-searches within the predefined geographical area. In the end, 13 organizations agreed to participate in the study. They include both national and international corporations from 8 fields of industry and public sector (see Table 1). The size of the organizations varied between 700 and 27 000 employees. The median of staff amount was 6 000.

Field of industry	**Amount of organizations**
Public administration	3
Industrial engineering	3
Retail	2
Education	1
Media	1
Health care	1
IT services	1
Finance	1
Total	**13**

Table 1. Distribution of organizations' fields of industry

Of the 13 organizations 15 persons were interviewed. The interviewees were high ranking IT officers, including CIOs and IT department directors (see Table 2), or other high level managers whose responsibilities included those of the CIO's (e.g. interviewee G).

Interviewee #	Interviewee title	Field of industry
A	Director of IT department	Health care
B	System specialist	Health care
C	CIO	Industrial engineering
D	CIO	Public administration
E	Corporate IT vice president	Industrial engineering
F	CIO	IT services
G	Development services manager	Education
H	IT manager	Media
I	CIO	Retail
J	CIO	Public administration
K	Head of IT department	Public administration
L	Head of IT department	Finance
M	CIO	Retail
N	IS manager	Public administration
O	CIO	Industrial engineering

Table 2. The interviewee profile

All interviewees were given the questions prior to the interview so that they could familiarize themselves with the topic. The actual interviews took place face-to-face at the interviewees' own premises. Interviews were digitally recorded with permission. Only the interviewer and the interviewee were present. In two cases, the CIO asked permission to bring an IT specialist (B and N) in the interview because they wanted to discuss the topic thoroughly. The permission was granted even though it caused dispersion to the interviewee profile by adding subjects that were not at the same organizational level as the CIOs. Data collection took place between February and April 2009. The recordings were transcribed in May 2009.

The structure of the interview consisted of background information questions and the actual interview question *"How would you define the life cycle of an information system?"*. Data was analysed against this question by clustering it into categories. The unit of analysis was information system. The clustering resulted three classes: 1) life cycle phases, 2) process integrated life cycle, and 3) life time. Each category reflects the perception the interviewees had towards the IS life cycle. This was followed by the interpretation of the CIO perceptions by the researcher. This was done by comparing the findings with life cycle theories and the use of life cycle concept in IS literature. This phase included returning to the transcribed interviews in order to form an understanding or theory of the possible *reasons* for the differences in perceptions related to the IS life cycle definition.

The conduct of the empirical research can be criticised for excluding the question of the definition of IS in the beginning of the interview. However, this could have had a negative influence on the responses by locking them to a particular definition. By leaving the actual IS definition unasked the importance of the definition emerged naturally of its own accord. Also, it formed the basis for the interpretation of the implicit, unarticulated IS definition between the lines – a method that the interpretative research is built upon as suggested by Walsham (1995). The following section reports the findings of the interview study which consists of the *perceptions* that the CIOs have towards the IS life cycle. The *interpretation* of these perceptions, as suggested by Walsham (1995) is given in the Interpretation section.

4. Findings

Three distinct ways of defining IS life cycle were identified from the data. Firstly, a majority (12) of the interviewees described the life cycle as a sequence of life phases. Secondly, a smaller, yet distinct portion of the interviewees (6) defined the IS life cycle as a continuously evolving system that is an inseparable part of the business or business processes. Thirdly, 5 interviewees defines the IS life cycle as the system's life time. Most interviewees brought up more than one viewpoint to the IS life cycle (see Table 3).

Interviewee #	Life cycle phases	Process integrated life cycle	Life time
A	x	x	x
B	x	x	
C	x		
D	x	x	x
E	x		x
F	x		
G	x		
H		x	
I	x	x	
J	x		
K	x		x
L	x		
M		x	
N			x
O	x		
Total	**12**	**6**	**5**

Table 3. Division of the IS life cycle definitions

Life cycle phases

The most common way of defining IS life cycle was by describing the sequence of stages an IS goes through during its life (see Table 4). Majority of the interviewees (12 out of 15 interviewees) associated the stages in some way with IS life cycle definition. The phases related to life cycle varied from "beginning and end" to ISD project phases, describing the process of defining, building, implementing, maintaining and disposing a system as stated by interviewee J: *"So we start with the basic definition of needs and the normal project proceeds in stages and when we launch the maintenance and support phase at some point […] I see the* *life cycle model from the very beginning to the point where again we move on to a new system, if we stop the use in the other end at some point."*

Interviewee #	Phases (extracted from the interviews)
A	Need, decision, building, use, maintenance, geriatric care
B	Feasibility study, decision to start a developmen project, requirements specification, acquisition or development in-house, implementation, maintenance, evaluation (continuous), evolution, giving up, replacement, changing the system
C	Specification of need, selection of system, system adaptation, implementation, upgrades, close-down
D	Specification, technical planning, transition into production, maintenance, development, continuous development
E	Identification of need, specification, version changes, upgrades, releases, geriatric care, putting down, data archiving, data conversions in a new system,
F	Need for change, need for more functionality, requirements collection from business and processess, development, implementation, use, updates, changes, further development, end of the road, close-down
G	Acquisition process: requirements specification, request for offers, selection, use (for ever), end (seldom)
I	Implementation, giving up, turning off
J	Normal project: Basic survey, specification of needs, trigger for development, maintenance and support, system change, end of use
K	Identification of needs, feasibility study, specification, technical specification, development, testing, transfer into production, maintenance, small-development phase, close-down, archiving, data conversions, turn off power
L	Idea and business need, selection: in-house building or acquisition, implementation, gaining business benefits, additional development needs, renewal needs, life cycle management phase (development, completion, fixing), system renewal, replacement
O	Idea, concept, requirements, development project, system's life on its own, does not correspond to reality, un-plug the system

Table 4. Life cycle phases

The findings suggest that the life cycle on an IS begins with the identification of need for change. *"It starts with the identification of need, feasibility study, definition, technical definition, building, testing, transfer to the production"* (Interviewee K). The trigger for change can be, for instance, that the existing system does not meet the current requirements. Next, is the feasibility study and decision making whether the project should be started or not. One of the CIOs pointed out that the *"the most important decision that relates to any information system life cycle is whether the system will be built at all"* (interviewee A). Requirements are then specified. System is acquired based on the specifications. This includes bidding and system selection. The selection takes place between alternative available products or between alternative vendor candidates system building as described by interviewee G *"So we do the requirements specification and then we proceed to the bidding stage and then we choose it"*. Next stage is the actual building of the system or acquisition project. The acquired or newly developed system is then implemented in the organization. Implementation is followed by use and maintenance as described by the head of IT department (K) "*Then there comes maintenance phase and this small-development stage [...] and then the close down, archiving and data conversions. It ends when the power is turned off and all necessary data has been transformed to some other format."*After some time the new requirements come up and the IS is modified accordingly. Evolution activities included system upgrading, version updates, complementary development, adding new features, and other changes. In the end, the system is closed down. It is replaced with another system or abandoned. These stages are described by interviewee L in the following way: *"Then additional development needs start to emerge, need to renew and we enter this so called life cycle management phase, during which the system is developed, complemented, repaired, etc, until it starts to be in a stage that either the business need has changed so much or then you have other business needs to renew the whole system and in that case you start with the idea that you will have a new system that replaces the previous one."*

There was some variation on how much emphasis was put on the stages. The activities related to the beginning and end of the life were emphasized over the middle stages (see Table 5). For instance use and maintenance were mentioned by three and four interviewees respectively. The identification of need, requirements specification and development or acquisition project were mentioned by eight interviewees. The most emphasized stage was system close-down. The issues related to the death of the system were discussed in detail by 10 interviewees.

Particular issues of concern were related to the end of life activities such as data conversion from the old system. The death was also described with diverse expressions including "*geriatric care*" (interviewees A and E), "*putting the system down*" (interviewee E), "*close-down*" (interviewees F and C) or "*unplugging the cables*" (interviewee O). The end was explained more than any other activity. There has to be a good reason for a close-down of any system. Reasons for close-down included: the system cannot support the business anymore, company buyouts where organizational merger that resulted multiple systems for the same function and excess systems are discarded, a system has degraded and has drifted beyond the point of cost-effective saving, legacy systems where qualified programmers no longer exist, vendor bankruptcy or vendor's decision to discontinue system development.

Phase	**Mentioned by interviewees**	**Total amount mentioned**
Identification of need	A, C, E, F, J, K, L, O	8
Decision	A, B, L	3
Requirements specification	B, G, C, D, E, F, G, K, O	8
Acquisition or development project	B, C, G, A, K, F, J, O	8
Implementation	B, C, D, F, I, K, L	7
Use	A, F, G	3
Maintenance	A, B, D, J	4
Further development (changes)	L, K, F, E, C, B	6
Closedown	A, B, C, E, F, I, J, K, L, O	10

Table 5. Summary of the life cycle phases

The interviewees' description of life cycle stages was typically complemented with a description of factors that triggered a shift from one stage to the next. Development or acquisition project is triggered by two types of need. Type 1 is the need to change the existing system. Type 2 is the need to add something new to an existing system. "*The thought arrives that now something has to be done, something is missing or something does not comply with needs* (interviewee F)" The need for change originates from three sources: business, technology, and vendor. The main change driver is business. It sets the framework in which the IS operates. According to a CIO from financing company, change is required when the "*business has changed*" Demands from the organisation's external environment, such as changes in legislation, regulate the business environment and subsequently influence the development of organisation's IS. Technology, on one hand, sets limitations and, on the other hand, guides evolution. Technology grows old and has to be replaced, or IS starts 'living its own life' and degrading. Vendors triggers change by giving up system support.

Process-integrated life cycle

The process-integrated IS life cycle class (see Table 6) consists of the answers that define IS life cycle through its functional role in an organization. IS life cycle is integrated to business processes. The interviewees emphasized that while the organization fulfils its task the IS has to adapt to any changes in it. IS is changed or renewed according to business needs.

Interviewee #	Process integrated life cycle (extracted from the interviews)
A	Basic organizational functions and processes remain unchanged. When processes change systems grow and change accordingly.
B	Evolution is significant, systems and their vendors change but the basic need does not dissappear.
D	Information system is a part of processes and services, is shares their life cycle. Systems barely ever die.
H	User interfaces change into more colourfull ones and external appearance changes but the basic philosophy, the purpose of use, does noe change. These is no end to the system life. It only changes shape during time and data is transferred from one database to another.
I	Systems exist and evolve within IS architecture. The functions of tasks, e.g. communications, exists even though its technical component has been changed many times, yet the basic functionality remains. As a functional unit, a system is difficult to define because it is connected to the process that utilises it.
M	The life cycle is defined by business. IS has to adapt to the changes in the business and change accordingly.

Table 6. The process integrated life cycle

Six interviewees (A, B, D, H, I, M) related the IS life cycle to their own organization's business processes. As an exception to this, one interviewee (G) stated that in their organization the IS life cycle is defined by the government's data administration policy. *"Then there are cases where they are dictated, that now everyone in the government office* *use this kind of [system] and we cannot really have any influence on them in any way. And, the life cycle then also depends on what the state dictates there at the background and the government data administration*" (interviewee G).

The IS was perceived as a part of a larger entity, integrated with various other systems and connected to a specific function (or functions) in an organization. The interviewees argued that an IS does not have a life span with beginning and end. Instead, they perceived IS as an inseparable part of organization's operations that continuously changes and evolves according to the organization's needs. The interviewees argued that the life cycle cannot be defined without the organisational context. The representatives of this category found defining IS life cycle challenging. They argued that IS life cycle follows the organisation's life cycle and changes when the business requires a change. Nevertheless, most of them recognised the life stages of a technical system but considered the process-integrated approach more relevant from business perspective than plain technical approach.

Besides the integration to organisational processes, inter-IS integration was recognised. A development and service manager from education field (Interviewee G) pondered this issue by saying: *"the life cycle is perhaps more and more related to the issue that they [IS] are not perceived as individual systems but how they are integrated with each other, and as a whole, a whole of systems. Then of course interfaces are needed, quite a lot of them, between different systems. So, maybe this way one could think about life cycle, I suppose."* IS architecture was brought up with system integrations. The interviewees considered it vital to recognise the relationships between different systems and see the 'big picture' of IS. The increasing system integrations caused concern about the manageability and individuality of the organization's IS. "*Besides lately I have also had fear that they start getting more* *complicated and so unreasonably difficult, these information systems, that soon it is not possible any more to renew a whole system starting from the beginning. Nobody can convert all the data from them into a new system and so one has to develop just certain parts of it. In a way the information system can seemingly stay in operation even though it has all been renewed little by little"* (Interviewee N).

Evolution was closely related to the process integrated life cycle. Two enablers of continuous evolution were found. The first enabler is the fixed basic need and stable organizational processes. A stable process base enables the use of the 'same IS' forever. The versions change, modifications are made, technology is renewed, and new features are added, yet the core functionalities remain the same. *"The evolution is an important issue. The systems are changed but the basic need does not go anywhere"* (interviewee B). The second evolution enabler is the emergence of new IS development approaches. Agile development methods (Nerur and Balijepally 2007) provide a system base that is easy to change and create the possibility for continuous improvement. In doing so it makes it unnecessary to terminate the system one day in the future. A CIO from the public sector (interviewee D) stated the following: *"So this has come to us, too, this agile system development. We are developing all the time and iterating and are trying to be a bit more dynamic than in the old days"*. Another evolution enabler, particularly for legacy systems, is Service Oriented Architecture (SOA) (see e.g. Sneed 2006). SOA extends the IS life time by allowing the old core systems to run underneath a new service layer untouched and enables system integrations on higher levels in system architecture (Sneed 2006). CIO from retail (interviewee M) said *"it [the old system] can fulfil the basic task and these certain interfaces are well defined, what can be retrieved from there and what can be stored there"*. New development methods also alter the consumer expectations when acquiring new IS. The IS is expected to be built to change. This becomes apparent from the following statement (interviewee G): *”it [IS] is selected with the assumption that they will be in use forever”*.

Life time

The concept age was related to the definition of IS life cycle (Table 7). The average system age varies between 1-30 years. Interviewee E said *“all this time period in between [beginning and end of life] can vary from one year to twenty years depending on the system”*. Interviewee A said *“The life cycle is fifteen years, or the time in use is fifteen years”*.

Interviewee #	System life time (years)
A	15
D	20-30
E	1-20
K	15-20

Table 7. System life time

Some interviewees made a difference between the total life time and version life time. For instance the total life time of an IS can be 20 years, while the life time of a system version is 5-10 years (interviewee O). One interviewee (N) defined life cycle as system life time in use: *“I myself consider that it [life cycle] is the system’s technologically feasible time in use.”* Otherwise system life time was associated with the stages view of IS life cycle (interviewees A, D, E and K) or with both stages and process-integrated view (interviewees A and D).

5. Interpretation of the findings

The findings suggest that the perception one has on the IS life cycle reflects the perception one has towards IS itself. *”It depends what we consider as information system“*(interviewee

B). This supports the proposition of Orlikowski and Iacono (2001) that the definition of the IS life cycle is dependent on the view the one has towards IS. This was stated by interviewee I: "*If we speak from technical point of view the system life cycle is easily defined. […][It is] when the technology is adopted and when it is given up. But if we speak about functionality, the system as a functional entity, then the it [the life cycle] is considerably more difficult to define. In my opinion, it is connected to the process that utilize it"*.

The interpretative analysis revealed two distinct ways how IS was viewed by the CIOs. IS was perceived as a *technical* system or as a *socio-technical* system. Viewed from technical perspective, an IS has has a clear beginning and end and as described by interviewee H: *"Hardware grows old or spare parts are not possible to get. They have to be changed therefore"*. Viewed from socio-technical perspective, the IS consists of IT, tasks, people, processes, and lives and develops continuously with the organization and the processes it supports as articulated by interviewee D: *"information system does not have a life cycle but instead it is the life cycle of the process and the service and the information system is just a parti in it, so the life cycle of an information system is similar to the actual process."*

This reveals that the technical view towards IS is related to the life view that has a beginning and end. Socio-technical view connects the IS to the human actors through processes (see e.g. Alter 2008, Lyytinen and Newman 2008). The findings provide basis for making an interpretation that the life time and life stages view towards IS life cycle are associated with technical view towards IS itself. Process-integrated IS life cycle view on the contrary is related to the socio-technical IS view (see Table 8).

Interviewee #	Socio-technical IS view	Technical IS view	
	Process integrated life cycle	Life cycle phases	Life time
A	x	x	x
B	x	x	
C		x	
D	x	x	x
E		x	x
F		x	
G		x	
H	x		
I	x	x	
J		x	
K		x	x
L		x	
M	x		
N			x
O		x	
Total	**6**	**12**	**5**

Table 8. Division of the IS life cycle definitions

6. Discussion

As a response to Orlikowski and Iacono's (2001) call for theorizing on IT artifact this paper states the following: Firstly, the IS life cycle has triple nature. The CIOs perceive it as a sequence of stages, process-integrated life cycle or life time. The definition of the IS life cycle is dependent on the view that the perceiver has towards the IS itself. Secondly, the IS definition has dual nature. On one hand it can be defined as a technical system, on the other hand IS can be defined as a socio-technical system. As stated by Orlikowski and Iacono (2001), the definition of IS depends on the perceptions of the perceiver. More specifically, these perceptions are related to the scope. Narrow scope, that is associated with technical IS definition, has closer system boundaries that rule out the organizational context. Wide scope, associated with IS definition as a socio-technical system, has system boundaries that include the organizational context. Based on these findings a question arises: Is the search for *one* IT artifact definition justified? Should the IS community accept the multiplicity of perceptions and continue with the perceiver-dependent IS definition instead of continuing the quest for a universal definition for IS?

Many organizations underestimate the issues and problems related to ERP life cycle (Chang et al. 2008). This study aims at increasing the understanding and awareness of the IS life cycle in organizations in order to reduce IS life cycle management related problems. The findings on process-integrated life cycle exhibit similarities with socio-technical models including Benbasat and Zmud's (2003) definition for IT artifact and Alter's (2003) work system. Surprisingly, the assumption that organizational IS life cycle follows WSLC model was not fully met. Instead, technical project model and life time based views were common. The life cycle phases view resembles the linear stage models in literature, e.g. ERP life cycle models of Esteves and Pastor (1999) and Chang et al. (2008), or IS development process (Avison and Fitzgerald 2003). Interestingly, system close-down received significantly more attention than other stages. This gives indication that close-down it is a critical event in practical IS management. This finding correlates to the high number of end-of-life studies of legacy systems (see e.g. Bennett et al. 1999, Bergey et al. 1997, Sneed 2005).

The technical perspective has advantages in the operational level IS management. It facilitates the conceptualisation and management of the IS because it can be divided into small and manageable stages. The disadvantage is that the "big picture" of the IS as a whole may get blurred. The life cycle phases view forms a base for project-oriented IS management where each stage is treated as a separate project. The life cycle is thus a sequence of projects: feasibility study project, acquisition project, implementation project, etc. IS as a socio-technical construct evolves continuously as the organization changes (Leavitt 1965). The life cycle of a socio-technical system is context-dependent as stated by Orlikowski and Iacono (2001). It is “IS and its circumstances” (see Ortega y Gasset 1914). This makes the IS life cycle management a much greater challenge that necessitates holistic solutions such as enterprise architecture (Lankhorst 2005).

The findings suggest that IS life cycle is a difficult concept to define. As a part of the socio-technical construct, the individuality of an IS is impugned. The measurement unit becomes clouded and system boundaries are difficult to define. Consequently, IS life cycle is difficult to define. IS life cycle definition depends on the viewpoint one adopts towards the IS itself. Also, it has a different meaning for different people. Hence, a shared agreement on the IS definition and an understanding the IS life cycle are prerequisites for managing the ISs in an organisation. It can be concluded, that the technical perspective is suitable for operative level IT management, socio-technical perspective is necessary for long-term strategic IS planning and overall IS management.

7. Conclusions

In this study, IS life cycle was studied theoretically and empirically from management point of view. CIOs from various fields of industry were interviewed in order to gain empirical evidence on the nature of IS life cycle in organizations. The objective of the interview was to encapsulate the perception that the respondents had towards IS life cycle. The findings suggest that CIOs perceive IS life cycle in three different ways: as a sequence of stages, as system life time or as a life cycle that it intergrated with organizations’ processes. Interpretation of these findings reveal the duality of the underlying IS definition. The life cycle of an IS as a technical system is associated with a sequence of life stages and system life time. The life cycle of an IS as a socio-technical system represents to the CIOs a system that is integrated in organization's processes and continuously evolves with them.

These views represent the two sides of a coin. Being seemingly different, yet they complement each other by representing two different perspectives on IS. The organizational life cycle frames the IS in large scale: it defines the beginning and the end of processes and work systems. This large scale life cycle on its turn consists of numerous evolutionary cycles, i.e. births and deaths of technical systems. Hence, it can be derived that an IS has innumerable evolutionary cycles within a life cycle which is defined by the organization. Systems and their versions may change but as long as the task or process remains the work system is alive. Replacement is the end of a system in a technical sense. Despite of a technical change the social system continues fulfilling its goals and the life cycle of a socio-technical system continues.

This paper has implications for both practitioners and researchers. The implication for practitioners is the potential difference of views that one may face when in IT management. The operative level IT management benefits mostly from adopting the technical perspective on IS life cycle while adoption of the socio-technical perspective is necessary for strategic IS planning and management. The implications for research is the increase of understanding of the variety of perceptions of IS life cycle in practice. The findings commit to the IT artifact discussion by providing empirical findings on how practitioners perceive the IT artifact and its life cycle.

The research was conducted in Finland. Care should be taken when generalizing the findings that are culturally bound. The uneven gender distribution of the interviewees is potentially limits the interpretations of the results. Fourteen out of the fifteen interviewees were males, leaving only one female. However, when considering the job description of a CIO, the gender distribution unlikely influences the results. The data somewhat reflects the economic depression and possibly added certain undertones to the results.

Further research is needed to explore the implications of the technical and socio-technical view of the CIO' towards IS life cycle management practices. Other suggested topics for further research include a historical investigation of life events of long-lived IS and the influence of outsourcing services such as SaaS (software as a service) on IS life cycle.

Acknowledgment

I am grateful to the [name removed for the blind review] for supporting this research financially. Also I would like to express my gratitude to the interviewees for the time they obliged for this research.

References

Alter S (2003) Sidestepping the IT Artifact, Scrapping the IS Silo, and Laying Claim to "Systems in Organizations". Communications of the Association for Information Systems 12, 494-526.

Alter S (2008a) Defining information systems as work systems: implications for the IS field. European Journal of Information Systems 17(5), 448-469.

Alter S (2008b) Service system fundamentals: work system, value chain, and life cycle. IBM Systems Journal 47(1), 71-85.

Avison D and Fitzgerald G (2003) Information systems development: Methodologies, Techniques and Tools, 3rd edition. Berkshire, McGraw-Hill Education.

Beatty RC and Williams CD (2006) ERP II: best practices for successfully implementing an ERP upgrade. Communications of the Association for Information Systems 49(3), 105-109.

Benbasat I and Zmud RW (2003) The Identity Crisis Within the IS Discipline: Defining and Communicating the Discipline's Core Properties. Management Information Systems Quarterly 27(2), 183-194.

Bennett K, Ramage M and Munro M (1999) Decision model for legacy systems. IEE Proceedings - Software 146(3), 153-159.

Bergey JK, Northrop LM and Smith DB (1997) Enterprise Framework for the Disciplined Evolution of Legacy Systems. Technical report, CMU/SEI-97.

Chang S, Yen CD, Huang S and Hung P (2008) An ERP System Life Cycle-Wide Management and Support Framework for Small- and Medium-Sized Companies. Communications of the Association for Information Systems 22(15), 275-294.

Davis AM, Bersoff EH and Comer ER (1988) A Strategy for Comparing Alternative Software Development Life Cycle Models. IEEE Transactions on Software Engineering 14(10), 1453-1461.

De Lucia A, Francese R, Scanniello G and Tortora G (2008) Developing legacy system migration methods and tools for technology transfer. Software: Practive and Experience 38(13), 1333-1364.

Denzin NK and Lincoln YS (2000) Introduction: The Dicipline and Practice of Qualitative Research. In Handbook of Qualitative Research (Denzin NK and Lincoln YS, Eds), California, Sage Publications.

Esteves JM and Pastor JA (1999) An ERP Life-cycle-based Research Agenda. In Proceedings of the First International workshop in Enterprise Management and Resource Planning: Methods, Tools and Architectures, pp. 359-371, Venice, Italy.

Fontana A and Frey JH (2000) Interview – From Structured Questions to Negotiated Text. In Handbook of Qualitative Research (Denzin NK and Lincoln YS, Eds), pp 645-672, Sage Publications, California.

Lankhorst M (2005) Enterprise Architecture at Work - Modelling, Communication, and Analysis. Springer-Verlag Berlin Heidelberg, Germany.

Lehman, MM. and Ramil JF (2003) Software evolution-Background, theory, practice, Information Processing Letters, Volume 88, Issues 1-2, 33-44.

Lyytinen K and Newman M (2008) Explaining information systems change: a punctuated socio-technical change model. European Journal of Information Systems 17, 589-613.

Markus ML and Tanis C (2000) The Enterprise System Experience — From Adoption to Success. In Framing the domains of IT management: Projecting the Future Through the Past (Zmud RW and Price MF, Eds), Cincinnati, Pinnaflex Educational Resources.

McGann ST and Lyytinen K (2008) The Improvisation Effect: A Case Study of User Improvisation and Its Effects on Information System Evolution. ICIS 2008 Proceedings, paper 209.

Mehra A and Seidmann A (2006) The Economics of Software Upgrades Throughout the Product Life Cycle. In the Proceedings of the 39th Hawaii International Conference on System Sciences(HICSS 2006), IEEE Computer Society, Washington, DC.

Miller D and Friesen PH (1984) A longitudinal study of the corporate life cycle. Management Science 30(10), 1161-1183.

Nerur S and Balijepally V (2007) Theoretical reflections on agile development methodologies. Communications of the Association for Information Systems 50(3), 79-83.

Nordheim S and Päivärinta T (2006) Implementing enterprise content management: from evolution through strategy to contradictions out-of-the-box. European Journal of Information Systems 15(6), 648-662.

Orlikowski WJ and Iacono SC (2001) Research Commentary: Desperately Seeking the "IT" in IT Research – A call to Theorizing the IT Artifact. Information Systems Research 12(2), 121-134.

Ortega y Gasset J (1914) Meditaciones del Quijote.

Porra J (1999) Colonial Systems. Information Systems Research 10(1), 38-69.

Rajlich TV and Bennett KH (2000) A Staged Model for the Software Life Cycle. Computer 33(7), 66-71.

Sneed HM (2005) An Incremental Approach to System Replacement and Integration. In Proceedings of the Ninth European Conference on Software Maintenance and Reengineering, 196- 206.

Sneed HM (2006) Integrating legacy software into a service oriented architecture. In Proceedings of the 10th European Conference on Software Maintenance and Reengineering.

Stake RE (2000) Case studies. In Handbook of qualitative research, 2nd edition, Denzin N.K. and Lincoln Y.S. (Eds.), Sage Publications.

Van de Ven AH and Poole MS (1995) Explaining development and change in organisations. Academy of Management Review 20(3), 510-540.

Van Fenema P, Koppius O and van Baalen P (2007) Implementing packaged enterprise software in multi-site firms: intensification of organizing and learning. European Journal of Information Systems 16(5), 584-598.

Varghese J (2003) ROI is not a formula, it is a responsibility. Journal of business strategy, 24 (3), 21-23.

Walsham G (1995) Interpretive Case Studies in IS Research: Nature and Method. European Journal of Information Systems 4(2), 74-81.

Walsham G (2006) Doing Interpretive Research. European Journal of Information Systems 15(3), 320-330.

III

LEGACY SYSTEM EVOLUTION – A COMPARATIVE STUDY OF MODERNISATION AND REPLACEMENT INITIATION FACTORS

by

Irja Kankaanpää, Päivi Tiihonen, Jarmo J. Ahonen, Jussi Koskinen, Tero Tilus, ja Henna Sivula

Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS 2007)

LEGACY SYSTEM EVOLUTION

A Comparative Study of Modernisation and Replacement Initiation Factors

Irja Kankaanpää
Information Technology Research Institute, University of Jyväskylä, Mattilanniemi 2, Jyväskylä, Finland
irja.kankaanpaa@titu.jyu.fi

Päivi Tiihonen
Jyväskylä University of Applied Science, Piippukatu 3, Jyväskylä, Finland
paivi.tiihonen@jamk.fi

Jarmo J. Ahonen
Department of Computer Science, University of Kuopio, Kuopio, Finland
jarmo.ahonen@uku.fi

Jussi Koskinen
Department of Computer Science and Information Systems, University of Jyväskylä, Jyväskylä, Finland
koskinen@cs.jyu.fi

Tero Tilus, Henna Sivula
tero@tilus.net, henna.sivula@gmail.com

Keywords: Legacy system evolution, evolution initiation, decision making, initiation factor.

Abstract: Decisions regarding information system evolution strategy become topical as the organisation's information systems age and start to approach the end of their life cycle. An interview study was conducted in order to compare factors influencing modernisation and replacement initiation. System age, obsolete technology and high operation or maintenance costs were identified as triggers for both modernisation and replacement projects. The results show that the most prevalent individual reason for modernisation initiative is business development. Common initiation factors for replacement projects were end of vendor support and system's inability to respond to organisation's business needs.

1 INTRODUCTION

Continuous evolution is necessary in order to maintain a system's ability to respond to the requirements of its environment (Lehman 1998). When a system ages and conventional maintenance faces increasing difficulties with system up-dates the organisation often faces a ”legacy dilemma” (Bennett 1995). A system with a long lifetime often contains accumulated business critical data yet preserving its functionality as such would require extensive use of resources and face difficulties due to obsolete technology (Bennett 1995).

There are three strategies to tackle a legacy system: 1) continuing maintenance despite of possible complications, 2) replacing the legacy system with a new system, or 3) modernising it (Bennett 1995, Seacord et al. 2003, p. 8-10). Modernization and replacement are significant economical investments with wide range of organisational effects. Consequently, evolution decisions are of great importance. In practice, the assessment of potential evolution options is challenging and decisions are often made informally and largely based on intuition (Saarelainen et al. 2006).

The aim of this study is to compare replacement and modernisation projects in order to provide an understanding of the factors influencing evolution initiation. A comparative study is important because when the differences and similarities of evolution types are known they can be taken into consideration in evolution planning and decision processes. The term evolution initiative refers to "a formally established and organized effort to evolve a system" (Bergey et al. 1997). Here, the term initiation factor refers to the decision basis and the reasons that trigger evolution activities.

2 EVOLUTION OPTIONS AND DECISION MAKING

In the past, several decision frameworks and models have been developed to systemise legacy system management and decision making. These include SABA – a decision model for legacy systems (Bennett et al. 1999), Renaissance (Warren and Ransom 2002), A decision framework for Legacy System Management (De Lucia et al. 2001), and Enterprise Framework for the Disciplined Evolution of Legacy Systems (Bergey et al. 1997). All these models emphasise the fact that legacy system decisions should include consideration of a number of factors and that the decision process should be rather formal in order to assure the success of system evolution. It has been proposed that the essential elements influencing legacy system evolution initiative are organisation, project, legacy system, target system, systems engineering, software engineering and technologies (Bergey et al. 1997).

Modernisation aims at improving the existing system to correspond with the requirements of its environment. It involves such radical modification that it cannot be considered as regular maintenance (Seacord et al. 2003, p. 9). Despite the significant improvements, a considerable proportion of the legacy system is conserved in the modernisation (Seacord et al. 2003, p. 9). Aversano et al. (2004) propose that prior to modernisation decision the technical and the business value of the existing system should be assessed. Koskinen et al. (2005) list twenty decision criteria influencing modernisation decision making, the ten most important criteria being: system usability, end of technological support, changes in business processes, maintenance costs, system correctness, system efficiency, expected remaining system lifetime, size of required changes, application domain expertise of maintainers, and delocalized system logic.

Replacement refers to an activity as a result of which a legacy system is replaced with another system providing the same functionalities (Warren and Ransom 2002). The replacing system can be purchased as an off-the-shelf application (Seacord 2003, p. 10) or it can be a result of a redevelopment project (Seacord 2003, p. 10, Bisbal et al. 1999). Bandor (2006) suggests that software selection process often focuses on the system functionality while ignoring the intangible factors related to the use and organisational influences of the system, and system management and operation related risks.

3 RESEARCH METHODS

The goal of the study was to compare two evolution types, i.e. modernisation and replacement, with respect to evolution initiation triggers. Therefore, it was relevant to collect empirical evidence on the topic. A qualitative research approach was chosen for it was desired to produce new information on the topic and not only to verify the results of previous research. For this purpose, semi-structured interview was considered as the most suitable data collection method (see Seaman 1999). Interview topics were selected and questions formulated based on a literature survey.

A set of pre-requirements were defined for acceptable projects for the interviews. The selection criteria were: 1) the project fitted either the modernisation or the replacement definition presented earlier, 2) the project was completed, and 3) a new or modernised system had been implemented in the organisation by the time of the interview. Also, an interview could relate to one evolution project only.

3.1 Data Acquisition

Data was acquired with phone interviews. Randomly picked organisations were contacted and inquired whether there had been suitable modernisation or replacement projects in the organisation's past (between years 2000 and 2005). If the response was positive, a key person involved in the project was contacted and requested an interview. The suitability of projects was assessed with the key person before the interview. If there had been more than one project that fitted the given criteria it was negotiated with the key person which project would be best suited to for discussion.

The person interviewed was most commonly the person with the deepest involvement or knowledge about the selected project. Prior to the interview, the interviewees were sent a list of the topics that were to be covered in the interview so that they could refresh their memory on the project and check documents or other data sources if necessary.

Data collection took place between autumn 2004 and autumn 2005. A total of 60 interviews were gained but all of them could not be used in the analysis for the following reasons: the project did not meet with the pre-requirements; or the project was a borderline case between the definitions of modernisation and replacement. These interviews were excluded from the analysis. The final material sample consists of 29 interviews from which 14 concerned modernisation and 15 replacement projects. Data was collected from 28 organisations, including both private companies and public organisations, and from 29 evolution projects within. One person was interviewed per project. One organisation yielded two suitable projects and thus the number of projects (and interviews) is higher than the number of organisations.

Most of the interviewees were upper or middle level managers in data administration. The sample also included IT development executives and other IT personnel. The study focused on user organisations and, consequently, system suppliers were excluded.

3.2 Data Analysis

Interviews were recorded by the permission of the interviewee and then transcribed. Data collection was closed when the material was saturated, i.e. new cases did not bring new information on the topic. In data analysis, classification by theme, type, and content breakdown were used. Before the analysis the material was read through several times in order to gain familiarity with it and to identify features that describe the material in general. The material was coded by theme in order to increase the fluency of handling. After the theme classification, the material was classified according to the frequency of response types in order to increase the comparability of the results.

3.3 Reliability of the Results

The reliability of the results was assured by using two researchers to analyse the material. They were not connected to the organisations and approached the material from an outsider's perspective. It was noted for that the reliability of the results could be weakened by the fact that the material was relatively heterogeneous, i.e. it consisted of different types and sizes of organisations and information systems. However, the responses were fairly homogenous in both evolution types. This indicates that the diversity in target organisations and systems have not undermined the reliability of the results.

The use of semi-structured interview provided a possibility to acquire in-depth information on the same topic from a variety of interviewees. The issues and viewpoints the interviewees brought up in addition to the pre-designed questions were relevant to the topic and added depth to the material. The reliability of the results could have been improved with additional quantitative data collection.

4 RESULTS

In the following, the results on evolution initiation comparison are reported. Citations[1] from the interviews are included in the text in order to clarify the results. The proportion of the appearance of each factor with respect to the total amount of projects is given in brackets (number of appearance / total amount). Abbreviation "R" is used for replacement and "M" for modernisation.

A total of 15 initiation factors were identified in the study. Some of these appeared in both modernisation and replacement projects, while some were evolution type specific. Factors influencing both modernisation and replacement initiation are system age, obsolete technology and high operation or maintenance costs. Identified replacement specific initiation factors were end of vendor support, need to unify disintegrated systems and scattered data, need for a system compatible with organisation's other systems, software or hardware, and to follow organisation's IT strategy or the prevailing IT trends within the industry. Modernisation specific initiation triggers were the desire to develop organisation's business or business processes, the requirements presented by a business

[1] Citations have been translated from Finnish to English by the author.

partner, responding to customer needs, legislation changes, and competitive advantage.

Table 1 summarises the initiation reasons and the number of their occurrence in replacement (column "R") and modernisation (column "M") projects. The number or occurrence presents the total number of projects where a presented factor appeared.

Table 1: Reasons for evolution initiation.

Reason for evolution initiation	**R**	**M**
System age	9	5
Obsolete technology	3	4
High maintenance or operation costs	1	2
Maintenance difficult or not possible	1	3
End of vendor support	7	-
Incompliance with business needs	7	-
Disintegrated systems and scattered data	3	-
Incompatibility with other systems	3	-
Incompliance with IT strategy	2	-
IT trends in the field of industry	1	-
Business or business process development	-	7
Customer needs	-	2
Business partner requirements	-	1
Change of legislation	-	1
Competitive advance	-	1

4.1 Replacement Initiation

The results suggest that replacement initiation is triggered by ten factors. These factors and their distribution throughout the replacement projects is depicted in Table 2, where R# denotes replacement project and x indicates the appearance of initiating factor in a project. Column T^R gives the total number of initiation factors' appearance across the replacement projects.

Three factors appeared significantly more frequently than the others: the system age (9/15), the end of vendor support (7/15), and the system's inability to respond to company's current business needs (7/15). Often these three major reasons appeared together or combined with other factors. (R7): *"The system was mainly from the year 1985 and then further developed. Its basic structure was nearly 20 years old … operations had changed significantly during that time and business requirements had changed so much that it was decided to renew the system."*

In almost half of the replacement projects (7/15), the reason for acquiring a new system was the legacy system supplier that had stopped or had announced the termination of system updates, maintenance or technical support. Typically, the vendor had informed the customer that development of the system would be discontinued. In R5, R11 and R12, the existing system was not compatible with other systems in its operational environment, with new operating systems or up-dated hardware. In the following, an interviewee (R5) describes a project where both of these factors were present: *"… we had a very old, about 10 years old, information system in use and we were in a situation that we could not really get updates for it and it was not compatible with these new operating systems any more. So in that situation we had to renew the system."*

In R4, R9 and R12, the legacy system consisted of interoperating sub-systems and problems with scattered data arose. The goal was to unify separate systems by replacing them with a single ERP system that would store all data in one location. An interviewee (R4) concludes: *"…when there are various small systems and their maintenance costs are fairly high we aimed at [getting them] within one system. On the other hand we were thinking about the transparency of customer data, and that we were transferring the same data in so many places, so that was one reason why we wanted to get rid of those separate systems."*

In addition to old age and disintegrated systems, IT trends in the field of industry encouraged system replacement (R12): *Well, it [system] was aged and it consisted of separate systems and at some point we should have changed it anyway … and what I have heard is that those [companies] who used the same systems, they have had quite a strong changing wave going on."*

Organisation's IT strategy guided the decision making in two cases. In R12, the system was completely rewritten in order to comply with new architecture and operating system requirements defined in the IT strategy. In R4, according to organisation's IT strategy all systems were to be united and, hence, individual systems were integrated into one organisation wide ERP system.

4.2 Modernisation Initiation

The results suggest that modernisation initiation is influenced by nine factors. From these three types of modernisation initiatives can be identified: business development driven modernisation, legacy system

The largest individual modernisation trigger was the desire to develop the organisation's business or business processes (7/14). It was typical that the weaknesses of the legacy system were acknowledged and action was taken in order to improve related operations.

Modernisation activities were considered as a natural part of organisation's operations and they took place in regular intervals (M11): *"This project,* *we called it 'the development of IT environment for better business process support'. … the whole IT environment and architecture was developed. It supports business processes and business objectives better. … one can say that it is an endless [process] but now [in this project] the basic improvements have been done."*

In one case, where modernisation was seen as an enabler of organisational development, it was anticipated to provide competitive advance. The motive for modernisation was to be the first organisation in the national markets to use new technology (M12): *"Yes, it has reference value if we are the first ones to do something."* In another case, the prevailing IT trends within the industry were regularly reviewed and assessed from business development perspective (M5):

"I belong to a national data administration group and we meet fairly regularly and discuss about the state of the art of software development in our line of business."

In four modernisation projects the initiative originated solely from business development motives (M4, M12) or as response to external pressure (M6, M14). In M6, legislation had changed and required rapid changes in the system. In M14, modernisation was initiated by the change request of an important business partner and responding to it was perceived as an opportunity for system improvement to better respond to customer needs and for business development.

The results indicate that legacy system related factors form the largest group of modernisation initiation reasons. Old age (6/14), obsolete technology (5/14) and impossible maintenance (3/14) were typically mentioned. Economical factors accompanied obsolete technology. High operation costs (2/14) were additional motivators. In extreme cases, old age (M3, M9) or obsolete technology (M7) was the only reason for modernisation.

The third identified modernisation initiation type includes both technical and business reasons: system was old and did not serve the user organisation as well as in the past (M5), system was old and did not respond to new customer needs (M1), maintenance was not possible and organisation wanted to develop its business operations by modernising the existing system (M11), or maintenance was nearly impossible due to obsolete technology and system

Table 2: Reasons for replacement initiation.

Initiation factors	**R1**	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**	**R9**	**R10**	**R11**	**R12**	**R13**	**R14**	**R15**	**T^R**
System age	x		x	x	x	x	x		x			x			x	9
Incompliance with business needs	x		x			x	x		x	x				x		7
End of vendor support	x				x			x	x			x		x	x	7
Obsolete technology								x			x		x			3
Incompatibility with other systems					x						x	x				3
Disintegrated systems and scattered data				x					x			x				3
Incompliance with IT strategy		x		x												2
IT trends in the field of industry												x				1
Difficult maintenance		x														1
High maintenance costs				x												1

Table 3: Reasons for modernisation initiation.

Initiating factor	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**	**M13**	**M14**	**T^M**
Business or business process development				x	x					x	x	x	x	x	7
System age	x		x		x				x						4
Obsolete technology		x					x	x					x		4
Maintenance not possible		x									x		x		3
High operation costs		x						x							2
Customer needs	x													x	2
Business partner requirements														x	1
Change of legislation						x									1
Competitive advance												x			1

was modernised in order to better supported business processes and objectives (M13).

The distribution of initiating factors throughout the modernisation projects is depicted in Table 3, where M# denotes modernisation project and x indicates the appearance of a factor in the left hand side column in a project. Column T^M gives the total number of initiation factors across the modernisation projects.

5 DISCUSSION

The reasons behind replacement and modernisation initiatives were noticeably similar but varied in nuances. Legacy system related factors proposed in earlier studies (see e.g. Aversano et al. 2004, Bergey et al. 1997), i.e. system age, obsolete technology, or difficult and costly maintenance, influenced evolution initiation in both project types.

A separating factor concerning evolution strategy selection was the degree of evolution initiative anticipation. Organisations that decided to modernise their legacy system wanted to develop their business processes. They were, in general, aware of IT trends in their business line and launched modernisation in order to develop their business. The interviewees were not directly asked about their organisation's business perspective, yet it was mentioned as an important factor in half of the interviews. Thus, strategic business planning as management activity (see Bergey et al. 1997) was a central element. On the contrary, interviewees from organisations that had chosen replacement as an evolution strategy did not express the desire for business development. Instead, replacement was a reaction to the vendor's announcement of ending system support or to the system's inability to support organisation's business operations. This would suggest that a system management related risk appeared real (see Bandor 2006) and caused action. It is proposed that modernisation took place as a result of proactive situation assessment while replacement was a reaction to a change in organisation's internal or external environment. The reasons behind this phenomenon require further research.

The results show that legacy system replacement and modernisation differ from each other with respect to vendor influence on evolution initiation. Nearly half of the replacement projects were initiated because the system supplier had given up system development or support. However, in modernisation projects, vendor's actions did not influence evolution initiation. It is concluded that organisations that decide to replace legacy systems are more dependent on system suppliers than organisations that choose modernisation.

When compared to the earlier findings, it can be noticed that there is clear divergence between initiation factors although similarities exist, too. Table 4 lists the decision criteria reported by Koskinen et al. (2005), which are in the text denoted with D, and the findings of this study, denoted with M^{IF} (modernisation initiation factors) and R^{IF} (replacement initiation factors) in a descending order of importance. An interesting finding was that system usability, which was listed as the most

Table 4: Comparison of evolution initiation criteria.

No.	**Modernization decision criteria (D) (Koskinen et al. 2005)**	**Modernisation initiation factors (M^{IF})**	**Replacement initiation factors (R^{IF})**
1.	System usability	Business development	System age
2.	End of technological support	System age	End of vendor support
3.	Changes in business processes	Obsolete technology	Incompliance with business needs
4.	Maintenance costs	Maintenance not possible	Obsolete technology
5.	System correctness	High maintenance costs	Disintegrated systems and scattered data
6.	System efficiency	Customer needs	Incompatibility with other systems
7.	Expected remaining system lifetime	Business partner requirements	Incompliance with IT strategy
8.	Size of required changes	Change of legislation	IT trends in field of industry
9.	Application domain expertise	Competitive advance	High maintenance costs
10.	Delocalized system logic		Maintenance difficult
…	…		
15.	System age		
16.	Changes in business environment		

important modernisation decision criteria, did not occur among the evolution initiation factors of found in this study. An explanation for this phenomenon was not found and would require further research.

Another observation is that the modernisation decision criteria reported by Koskinen et al. (2005) correspond to replacement initiation factors better than modernisation triggers found in this study. Correlation for all but one replacement initiation factor, i.e. end of vendor support ($R^{IF}2$), exists: $R^{IF}1$ = D15, $R^{IF}3$ and $R^{IF}7$ = D3, $R^{IF}4$ = D2, $R^{IF}5$ and $R^{IF}6$ = D5, $R^{IF}8$ = D16, $R^{IF}9$ = D4, and $R^{IF}10$ = D7. $R^{IF}5$ and $R^{IF}6$ correspond to system efficiency (D6) because they led to inefficient system use in studied organisations. It should be noted that D2 refers to general technology changes and, hence, it should not be mixed with $R^{IF}2$.

Six of the identified modernisation initiation factors match with the previously presented criteria. The following correlations exist: D2 = $M^{IF}3$, D4 = $M^{IF}5$, D7 = $M^{IF}4$, D15 = $M^{IF}2$, and D16 = $M^{IF}6$, $M^{IF}7$. New factors, not appearing in the list by Koskinen et al. (2005), are business or business process development objective ($M^{IF}1$), end of vendor support ($R^{IF}2$), and competitive advance ($M^{IF}10$).

The amount of factors influencing evolution initiation decision is significantly smaller than given before (see Koskinen et al. 2005). The results strengthen the previous findings of the importance of engaging both business and technical aspects in evolution decision making (see Aversano et al. 2004, Bergey et al. 1997).

6 SUMMARY

The aim of this study was to compare replacement and modernisation projects with respect to the reasons that initiate evolution activities in order to provide evidence on their differences and similarities to support evolution planning and decision processes. The differences and similarities were mapped with an empirical study where data administration managers, IT development executives and other IT personnel from 29 evolution projects were interviewed.

The results confirm the previously suggested modernisation criteria but challenge the order of their importance. The findings question the importance of system usability, previously claimed to the most important modernisation decision criteria. New factors complementing earlier findings are business or business process development objective, end of vendor support, and gaining competitive advantage.

System age, obsolete technology and high operation or maintenance costs were identified as triggers in both evolution types. The most common initiation factors in replacement projects were system age, end of vendor support, and system's incompliance with the organisation's business needs. The most common reasons for modernisation were the desire to develop organisation's business or business processes, system age, and obsolete technology.

ACKNOWLEDGEMENTS

This work has been carried out in ITRI (Information Technology Research Institute) and financially supported by TEKES (Finnish Funding Agency for Technology and Innovation).

REFERENCES

Aversano, L., Esposito, R., Mallardo, T. and Tortorella, M., 2004. Supporting Decisions on the Adoption of Re-engineering Technologies. In *CSMR'04, 8ht European Conference on Software Maintenance and Reengineering*. IEEE Computer Society, 95-104

Bandor, M. S., 2006. *Quantitative Methods for Software Selection and Evaluation, Technical Note*, CMU/SEI-2006-TN-026. Retrieved November 15, 2006 from Carnegie Mellon University, Software Engineering Institute Web site: http://www.sei.cmu.edu/publications/documents/06.reports/06tn026.html

Bennett, K., 1995. Legacy Systems: Coping with Success. *IEEE Software*, 12 (1), 19-23.

Bennett, K., Ramage, M. and Munro, M., 1999. Decision Model for Legacy Systems. IEE Proceedings - Software 146 (3), 153-159.

Bergey, J. K., Northrop, L. M. and Smith, D. B., 1997. *Enterprise Framework for the Disciplined Evolution of Legacy Systems*, Technical Report, CMU/SEI-97-TR-007, ESC-TR-97-007, Retrieved November 15, 2006 from Carnegie Mellon University, Software Engineering Institute Web site: http://www.sei.cmu.edu/publications/documents/97.reports/97tr007/97tr007abstract.html

Bisbal, J., Lawless, D., Wu, B. and Grimson J., 1999. Legacy Information Systems: Issues and Directions. *IEEE Software*, 16 (5), 103-111.

De Lucia, A., Fasolino, A.R. and Pompella, E., 2001. A Decisional Framework for Legacy System Management, In *ICSM'01, International Conference on Software Maintenance*, IEEE, 642-651.

Koskinen, J., Ahonen, J.J., Sivula, H., Tilus, T., Lintinen, H. and Kankaanpää, I., 2005. Software Modernization

Decision Criteria: An Empirical Study, In *CSMR'05, 9th European Conference on Software Maintenance and Reengineering*. IEEE Computer Society, 324-331.

Lehman, M. M., Perry, D. E. and Ramil, J. F., 1998. Implications of Evolution Metrics on Software Maintenance. In *International Conference on Software. Maintenance*. IEEE Computer Society, 208-217.

Saarelainen M., Ahonen, J.J., Lintinen, H., Koskinen, J., Kankaanpää, I., Sivula, H., Juutilainen, P. and Tilus, T., 2006. Software Modernization and Replacement Decision Making in Industry: A Qualitative Study. In EASE'06, *10th International Conference on Evaluation and Assessment in Software Engineering*. Kitchenham, B.A., Brereton, P., Turner, M., and Charters, S. (eds.), Retrieved December 5, 2006 from BCS eWIC database.

Seacord, R. C., Plakosh, D. and Lewis G. A., 2003. *Modernizing Legacy Systems – Software Technologies, Engineering Processes and Business Practices*. Addison-Wesley, Boston, MA.

Seaman, C. B., 1999. Qualitative Methods in Empirical Studies of Software Engineering, *IEEE Transactions on Software Engineering*, 25 (4), 557-572.

Warren, I. and Ransom, J., 2002. Renaissance: A Method to Support Software System Evolution. In *COMPSAC'02, 26th Annual International Computer Software and Applications Conference*. IEEE, 415-420.

IV

TIMING THE INFORMATION SYSTEM UPGRADE

by

Irja Kankaanpää & Samuli Pekkola

The 18th European Conference on Information Systems (ECIS2010)

18th European Conference on Information Systems

TIMING THE INFORMATION SYSTEM UPGRADE

Journal:	*18th European Conference on Information Systems*
Manuscript ID:	ECIS2010-0338.R1
Submission Type:	Research Paper
Keyword:	CIO, Decision making/makers, Enterprise resource planning, IS life cycle activities

TIMING THE INFORMATION SYSTEM UPGRADE

Kankaanpää, Irja, University of Jyväskylä, PO Box 35 (Agora), 40014 University of Jyväskylä, Finland, irja.k.kankaanpaa@jyu.fi

Pekkola, Samuli, Tampere University of Technology, PO Box 541, 33101 Tampere, Finland, samuli.pekkola@tut.fi

Abstract

A system upgrade requires careful planning as its implications to organizational systems might be enormous. Although in IS literature the requirements and process of systems upgrade have been discussed, the timing when to upgrade and what factors guide it has been of lesser interest. Consequently, in this paper we focus on information systems upgrading and its timing from the perspective of the user organization. Upgrading is enabled by the availability of a new software version. When to upgrade, meanwhile, is determined by the business interests of the customer organization, business calendar, development projects, and the vendor. These factors were identified by interviewing 14 IT managers, mainly CIOs, from middle size to large organizations in Finland. They presented 16 different cases of upgrading or modifications of enterprise systems or similar undertakings. The analysis of the cases and the identification of the upgrade timing factors not only increase our understanding of the phenomena in general, but also reveal the customer's motives and interests regarding IS upgrading and its timing.

Keywords: Version change, system upgrade timing, vendor, customer, motives, determinants

1 INTRODUCTION

System upgrade is a decision making point for both the system vendor and the system user, i.e. the customer. In order to stay in markets, the vendors have to produce and sell new versions of their products. In fact, they are producing new versions faster than the customers can adopt them (Sawyer 2001). The system users think differently. They balance between acquiring the newest version and using their current system. As the enterprise resource planning (ERP) system implementation process is complex (Al-Mudimigh et al. 2001) and costly (Chen 2001, Stefanou 2001), the users have to consider both the fit between the system and their organizational needs (Hong and Kim 2002) and the timing of acquisition or upgrade. There is, consequently, a conflict of interest between the vendor and the customer in switching from one system version to another, i.e. upgrading the system.

ERP system implementation engages the customer in a tight relationship with the system vendor for a relatively long period of time (Markus & Tanis 2000, Verville et al. 2005). This kind of long-term marriage emphasizes the need for shared interests and mutual trust as they influence the quality of co-operation between the partners. Consequently, at the point of making decisions, the partners' motives and interests should be considered as they put the stakeholders in a position where their interests might conflict – if not immediately, then potentially over time.

Although system upgrades have become an integral part of IS management, the timing of a version change is unsystematically defined in the user organizations (Mukherji et al. 2006). Similarly, the investments to new systems have dominated IS research while the system upgrade has been overlooked (Mukherji et al. 2006). Particularly the timing of the packaged system upgrade has been a neglected in ERP system research (Ng 2001).

This paper studies the determinants for the timing of enterprise system (ES) upgrades, particularly ERP system upgrades, from the user organization's point of view. In other words, we aim at recognizing conditions that define system upgrade timing. The literature analysis focuses on clarifying the differing perspectives the user organization and the vendor have towards software product's life cycle. Also, the ES system evolution and vendor-customer relationship are analyzed. The purpose is to identify system upgrade related life cycle phases and to show how the differing viewpoints towards software product's life cycle generate differing, and partly conflicting, interests with respect to IS upgrade timing and frequency. This analysis has practical implications as it will help vendors and customers to understand the causes for potential conflicts with respect to IS upgrading by laying out the interests and motives of both parties.

In order to understand the customers' dynamics and determinants for ES upgrade timing, empirical research was carried out. 14 high level IT executives (mainly CIOs) were interviewed from 12 middle size to large Finnish companies and public organizations. The interviews yielded 16 cases concerning IS upgrading. Empirical findings provide in-sights on the drivers and inhibitors of system upgrade timing. They will help one in practice to schedule upgrading and understanding the required conditions.

The remainder of the paper is organised as follows. The next two sections discuss the software product's life cycle from both vendor and user organization's perspective and system upgrade and its timing. Research methods and settings of the empirical research are then described. This is followed by the reporting of the findings. The findings are further discussed and a model for the determinants of IS upgrade timing is presented. The paper ends with a summary.

2 CUSTOMER, VENDOR AND PRODUCT LIFE CYCLE

Enterprise systems are specialized and increasingly diverse (Messerschmitt & Szyperski 2003). Their development is driven by both the customer's and the vendor's interests and motives (Sawyer 2001). The vendor is interested in developing software products for the markets while the customer's interest is in assembling coherent IS out of software pieces (Sawyer 2001). The distinction between these perspectives is significant as they provide reciprocal, mutually complementary view towards an interesting phenomenon: timing the upgrade.

An organization that purchases an ES or ERP system enters into long-term relationships with the software vendor, even from five to ten years (Verville et al. 2005). This underlines the importance of the quality of the relationship between the customer and the vendor. However, due to the divergent perceptions towards the IS life cycle, potential conflicts of interest exist. The customer is dependent on the vendor for both system support and changes (Markus & Tanis 2000). This urges for an analysis of the software product's life cycle in order to be able to understand the relationship between systems development and its upgrading, the former performed by the vendor, the latter by the customer.

2.1 Software product's life cycle

The vendor's business is to sell software and related services. Software development process consists of five distinct phases: initial development, evolution, servicing, phase-out and close-down (Rajlich & Bennett 2000). Initial development refers to a phase where the system's first functioning version is built from the scratch. As initial development is expensive and risky with potentially no return, the vendor attempts to release the products as fast as possible in order to "generate revenue and to beat any competition" (Rajlich & Bennett 2000). Particularly at the beginning of the product's life cycle, the monopolist software producers use the tactics of "leveraging incompatibility" between different versions in order to force the customers to update their software (Mehra & Seidmann 2006).

The system seldom remains unchanged over time but instead tends to evolve. The "capabilities and functionality of the system" are extended to meet the changing needs of the users. The system undergoes a sequence of iterative changes triggered by customer demands, competitive pressure or legislative actions (Rajlich & Bennett 2000). The evolution is manifested through versions.

New versions are released at certain intervals until the sales drop. Software reaches the servicing phase (Rajlich & Bennett 2000). New versions are not assembled any more as the evolution becomes increasingly difficult or expensive. Under these circumstances the vendor provides corrective maintenance, that is, service patches. Service patches easily degrade the system architecture and speed up software decay which eventually leads to phase-out phase (Rajlich & Bennett 2000).

At the phase-out phase, the vendor tries to get revenue from the system with no efforts (changes, service patches) as long as possible before withdrawing the product completely from the markets. Since the servicing is stopped, the users have to work their way around the upcoming difficulties (Rajlich & Bennett 2000). Finally the system is closed down. The vendor withdraws it from the markets and induces the users to replace it by another system.

2.2 Customer's interests

The customer creates an IS from the available commercial software products (Sawyer 2001). These pre-fabricated products and their various versions are building material that are transformed into an operating and organizationally integrated IS by means of adoption and implementation. This process has been depicted by various life cycle models of ERP and ES systems (e.g. Esteves & Pastor 1999, Cooper & Zmud 1990, Markus and Tanis 2000).

Customer's IS related investment decisions and procurement strategies are largely guided by the markets (Narasimhan et al. 2006). For each system upgrade the company attempts to increase its compliance and alignment with their business needs. The customer is interested in the quality of the product, its delivery, and the vendor's responsiveness and innovativeness in order to evaluate how effective the vendor is and will be in meeting the customers varying needs (Narasimhan et al. 2006). Usually the customers want to be able to contribute to changes in the system. They also expect the new versions to be better aligned with their needs.

Because of the complexity and costs of the implementation of large IS (Al-Mudimigh et al. 2001, Chen 2001, Stefanou 2001), it is in the company's interest to upgrade systems only when needed. Frequent or unnecessary upgrades, even when being technologically feasible, cause problems by consuming staff resources and IT budget and by disturbing the end-users (Esteves & Pastor 2005, Mukherji et al. 2006). Consequently the customer wants to get the version changes done as rarely and efficiently as possible.

2.3 Conflicting interests between customer and vendor

The examination of the perceptions of the customers and vendors' IS life cycles reveals potential conflicting interests related to upgrade timing. These are summarized in Table 1. The software product life cycle from vendor's point of view has been taken as a reference model because the vendor's dominant position in the customer-vendor relationship.

Phase	Vendor's interest	Customer's interest
Initial development	Deliver a new product on the markets as fast as possible	Purchase a mature and reliable, fully tested software
Evolution	Offer new versions frequently	Upgrade only when needed and only versions with optimal functionally
	Meet with customer demands, competitive pressure, changes in legislation	Get custom changes
Servicing	Offer service patches instead of new versions	Get large scale changes or a new version
Phase-out	Provide no changes	Get changes for the system
Closedown	Stop providing system support	Continue the use of the system

Table 1. Potential conflicts of interests between vendor and customer.

At initial development phase, the vendor seeks to deliver a product on the markets as fast as possible while the customer is seeking for mature product with no "childhood diseases". Thus, a conflict arises from supply and demand not meeting at product quality and maturity level. At evolution phase, vendor's eagerness to offer new versions at high frequency (Sawyer 2001) and induce the customer to commit to a specific upgrade program conflicts with the customer's interest to upgrade systems only when needed. At servicing phase, conflicts arise from customer's change requests which the vendor is not willing or able to provide. An apparent conflict emerges at phase-out stage when the customer requests system support that is no longer provided.

3 DEFINING SYSTEM UPGRADE TIMING

Software products typically evolve fast and in small incremental steps. As new versions are released regularly, the customer's question is not 'whether' to upgrade the system, but when to do so (Mukherji et al. 2006). Particularly in the context of ERP systems, the users do not typically invest in every new version but instead "leapfrog to adopting a subsequent release" (Mukherji et al. 2006). Hence, the upgrade decision is dependent on the available versions and contingent factors. Legal issues may force (Mehra & Seidman 2006), lack of vendor support may push and potential competitive advantage may motivate the version change (Kankaanpää et al. 2007). However, financial issues play a major role in upgrade decisions.

System upgrades are less expensive and are expected to yield less benefit than investments to new systems (Mukherji et al. 2006). An upgrade still remains a considerable investment that requires joint organizational efforts. Frequent upgrades are both costly and risky, yet delaying the upgrade decision may "lead to loss of competitiveness" (Mukherji et al. 2006). However, postponing the upgrading increases the gap between the version in use and the supported version in markets. The upgrading becomes increasingly difficult as the gap grows (Mukherji et al. 2006). This emphasizes the importance of timing of the upgrade, which further necessitates the understanding of the vendor's product life cycle and developed plans (Narasimhan et al. 2006).

An economics-based decision model for defining the optimal IT upgrade timing (Mukherji et al. 2006) indicates that "investments in upgrades are best made when the gap between new technology and current technology reaches a critical threshold. Among other factors, this threshold is influenced by technology cost, change management cost and opportunity cost." (ibid). Technology cost refers to the costs of adopting a new version. Change management cost is the cost of upgrade deployment activities, e.g. the time used for learning new routines and re-training the users. Opportunity cost means the cost of lost opportunity. It can manifest itself through the decreased productivity or the loss of revenue due to the decision not to adopt new technology or to adopt an ill-fitting version.

The model suggests that in order to optimize the upgrade timing, the customer should wait until the technology cost of the new version decreases since the latest version is the most expensive immediately after its release. Also, if the estimated change management costs are high, upgrade should be postponed. However, if a company does not invest in new technology, it may loose the opportunity for higher productivity. (Mukherji et al. 2006). An optimal time for the upgrade occurs when the difference between the level of technology in use and the level of the most suitable new technology level hits a critical threshold. This means the company has to compare the technology and change management costs to the opportunity costs. "Leapfrogging" as an upgrade technique supports these findings: the most economical way of upgrading is to wait until there is a need for change, skip unnecessary versions, and upgrade when a clearly beneficial version is available. (Mukherji et al. 2006).

4 RESEARCH METHODS AND SETTINGS

In order to identify the preconditions influencing the timing of the system upgrade from the customer's point of view, 14 IT executives, mainly CIOs, from 12 mid to large size companies and public organizations in Finland were interviewed and are included here (Fontana and Frey 2000)[1]. Table 2 portrays the cases and their fields of industry.

Case #	Field of industry	Case description
A	Engineering	ERP system upgrade
B	Financing	ES system version change
C	IT services	ES system version change
D	Public administration	ES system vendor and version change
E	Public administration	ES system upgrade
F	Public administration	ES system upgrade
G	Retail	ES system version change
H	Public administration	Workstation and operating system upgrade
I	Retail	Workstation and operating system upgrade
J	Engineering	Groupware system upgrade
K	IT services	Invoice handling system version change
L	Retail	Accounting system change
M	IT services	Patient management system change
N	Engineering	Maintenance service system upgrade
O	Media	Publishing system change
P	Public administration	Intranet technology upgrade and vendor change

Table 2. Summary of the cases.

[1] An invitation was emailed to 37 organizations, from which 13 responded. This resulted 15 interviews: Four interviews yielded 2 IS upgrade cases, eleven interviews yielded 1 case, and one yielded no case. Only the interviews with the cases are included here. Consequently, the interviews yielded a total of 16 cases from 14 subjects representing 12 organizations.

The interview questions were formulated according to a literature survey (c.f. Kankaanpää & Maaranen 2009). A large number of detailed questions were drawn from the theory basis after which their abstraction level was raised. This decreased the number of questions. They were then pilot tested by three colleagues. The questions were revised according to the feedback before the actual interviews. Data collection took place from February to April 2009. The questions were emailed to the subjects prior to the interview. Each CIO was interviewed face-to-face in their premises – except in two sessions, where both CIO and his/her aide were present. Each interview was recorded with permissions, and transcribed for analyses.

For this paper, the data was analyzed against the interview questions (see Appendix A). Then, it was categorized according to themes that arose from the material. The researcher familiarized herself with the material thoroughly in order to gain deep understanding of the cases.

5 FINDINGS

The interviews show that the rationale for IS upgrade timing depends on business benefits, business calendar, organization's ongoing and planned development projects, and vendor. In the following, those are discussed in detail.

5.1 Business interests

The potential value for business, minimized hindrance for the business, and the organization's readiness for an upgrade provide a basis for IS upgrade timing. In general, these business related issues define the large scale time frame for upgrades.

Expected business benefits greatly influence the timing of the version change. As one of the CIO's described: "*the upgrade timing depends precisely on the value of its business case*[2]" (case B). This points out the customer's interest to evaluate the potential opportunities and added business value that could be reached with the new version. Likely business benefits encourage for rapid upgrade.

Despite the expected business benefits, the organization's financial situation has a word whether to upgrade. "*We had a good financial situation in our company. We have money to invest. And we can get a positive investment decision. That set the schedule*" (case J). However, another CIO provided slightly different viewpoint: "*So in this case the timing was chosen on the basis of when there are resources available for the version change. However, as a matter of fact, we upgraded the system as we had to do so*" (case B). The availability of resources and favorable financial situation are preconditions for a version change. When these conditions are met, the investment proposals can be more easily accepted and funds granted. Hence the timing of the version change of large IS, such as ES systems, is commonly decided in conjunction with planning and budgeting.

The business interests of the customer necessitate minimizing the hindrance for the business. Particularly this appears with non-business critical systems upgrades: *"Because this is not clearly business driven version change, it is timed in a way that it causes the minimum amount of inconveniences. If it was a business driven version change, then we would have defined the timing by the business case.*" (case B).

5.2 Business calendar

As simply as it is, possible timeslots for upgrading are defined by the organization's business calendar. Business calendar defines the annual business activities of a company. Business calendar is strongly

[2] Translations from Finnish are made by the authors.

influenced by the (national) accounting practices, laws, and cultural issues, such as people preferences and holidays. Also, the level of internationalization and industry specific business characteristics, define each organization's own business calendar. For instance, a CIO from retail industry described the influence of operating in multiple countries: "*the more we expand our business abroad, the smaller are the timeframes [for system upgrades]. There is no time in a day because the stores are open. This is a retail industry specific issue. The stores open up in different counties at different times, typically at 9 a.m., and then they close at 9 or 10 p.m. But 9 a.m. in another country is different than in Finland or somewhere else.*" (case L).

Each field of industry has its own high-business and low-business seasons. As a rule of thumb, IS upgrades are preferably made during the low-business seasons where the potential hindrance for business is at lowest. However, such windows for change are sometimes rare – as the quote from retail industry (case L) above points out. The changes can be made only outside the business hours, putting an emphasis on the quality of upgrading work. If the upgrade of a system fails, the next opportunity might be open a year later. Yet the businesses are not the same: for building sector and agricultural field the low-business season is winter, in public administration low-business seasons are March and October (no budgeting or reporting taking place), and for health care, there is no low season. *"Often they [upgrades] are related to system down-time and similar issues. When the health-care services rely on the system, the timing of a shut down is extremely important. And it cannot be down for many days. Yes, the timing is extremely important."* (case E).

Nevertheless, there are also several universal times when IS upgrades are avoided. Those include Christmas, turn of the year, and summer holiday season. The turn of the year demarks the end of the fiscal year and the beginning of the reporting period. The closing of the accounts and annual reporting cannot be compromised by any IS related risk. *"At the turn of the year, we always tried to quiet it down. The reason is that the turn of the year is critical for the banks. That is the reason. Then another thing is, that then there are usually lots of changes to taxation and such. And the reports are done and everything else. And when you have implemented new applications, new features etc. throughout the year, annual reporting, which is exceptional, occurs for the first time. So you aim at calming it down, you don't do much in that month. We set the close season from half way of December and it continues over the turn of the year."* (case B).

Similarly, during the summer season, or Easter time, the changes in IS are avoided. This is because the employees are on holiday, not available for doing upgrades. This means that factually there are two periods in a year when the upgrades are possible: "*From mid August to late November and from February to May, that's when the major changes are made*" (case N). Yet, these time frames are not continuous. Fiscal year and fiscal periods determine time frames for the possible IS upgrades as, for instance, reporting has to take place regularly. Optimally the best time for IS version change is at the end a fiscal year, immediately after the annual financial statements and reports have been published. This means that new versions should be deployed at the beginning of the year.

Different kinds of systems have different windows of opportunity for their upgrades. Accounting systems, for instance, have strict rules when any change can be made. This means that financial administration systems are frozen during the closing of the books in January and February. Also the connections between systems have to be considered as "*everything influences everything*" (case O).

5.3 Development projects

Systems are not upgraded in a vacuum. An upgrade is usually conducted in a project that is dependent on the organization's project calendar and development agenda, and internal policies and plans. For instance, an organization's IT strategy may define a frequency of system upgrades. This was the case in case C where ES upgrade was performed regularly, in every 2 years. In case H, workstations were updated in systemic 4-5 year intervals. The size of IS upgrade project also influences the timing. The bigger the project, the earlier it has to be planned and scheduled.

Ongoing and forthcoming development projects influence IS upgrade timing. Basically there are two strategies: to avoid concurrent projects (i.e. IS upgrade and other projects), or to integrate process improvement projects with an IS upgrade project. If simultaneous yet independent changes cause unnecessary inconveniences and hindrance to end-users, concurrent changes and overlapping projects are avoided. For instance in case J, as upgrading the groupware system changed the work practices of thousands of employees, the version change was timed carefully in order to avoid encumbering the end-users with other simultaneous projects.

The approach of integrating process improvement with IS upgrade takes an opposite approach. The rationale is to pack multiple changes into one big project. This means that the end-users have to go through a fundamental change in a short period of time instead of a sequence of smaller changes in long period of time. This minimizes the amount of hindrance to one's work and productivity. For instance in cases A and O, the timing of a version change were settled according to the business process development so that they would support each other, and help the users to adapt to multiple changes at the same time.

5.4 Vendor

The upgrade is evidently dependent on the availability of a new version. Sometimes the latest version is not fully operational or contains many bugs. The timing is thus dependent on the vendor's ability to deliver and guarantee a fully functional and usable version. Any uncertainty there leads to assuring actions by the customer. For instance in case I, the schedule of an operating system release was not clear. This forced the organization to update their workstations not with latest but an older version in order to avoid risks related to the newest and most desirable version.

Support services and their termination is a compulsive reason for upgrade. In case K, regardless of the vendor's suggestion, the organization had postponed their version change for several years. The vendor was reluctant to continue the support service despite of the organization's continuous requests. Finally, the vendor announced that the support ends and that they cannot take any responsibility for the system any longer. This defined the timing of the version change in a concrete fashion.

Case K also points out that vendor's interests are not always in line with customer's interests. The vendors propose version changes and upgrades from their own business perspectives and schedules which do not necessarily match with the customers' interest, needs or business calendar. Although some organizations found the vendor's upgrade programs useful and beneficial, in most of the cases, the conflicting interests between the vendor and the customer were experienced. Usually the vendor wanted to do changes that the customer did not want – or vice versa. "*Then there're the vendor's reasons why it would be good to do the version change now. Those reasons seldom match with ours. They usually don't. In other words, the vendor's desires always contradict with our model. Or very often they do. Nevertheless, the customer anyway understands that they have to change the version in order to keep up with the development, so how it should be scheduled then?*" (Case L).

No matter whether the vendor and the customer negotiate when to change the version, there are situations when the upgrade is not possible when wanted. For example in case D, version change was defined by the ending of the contract with the vendor. As soon as the contract ceased, the customer was able to change the vendor and upgrade the ES system.

6 DISCUSSION

The literature analysis suggests that the timing of the IS upgrade is dependent on the availability of a suitable version, the customer's need for upgrade and economics. These are supported by the empirical findings of this study. Additionally, the empirical findings suggest that, from the customer's point of view, the timing of an IS upgrade is guided by four determinants, namely business interests, business calendar, on-going and planned development projects, and the vendor (see Figure 1).

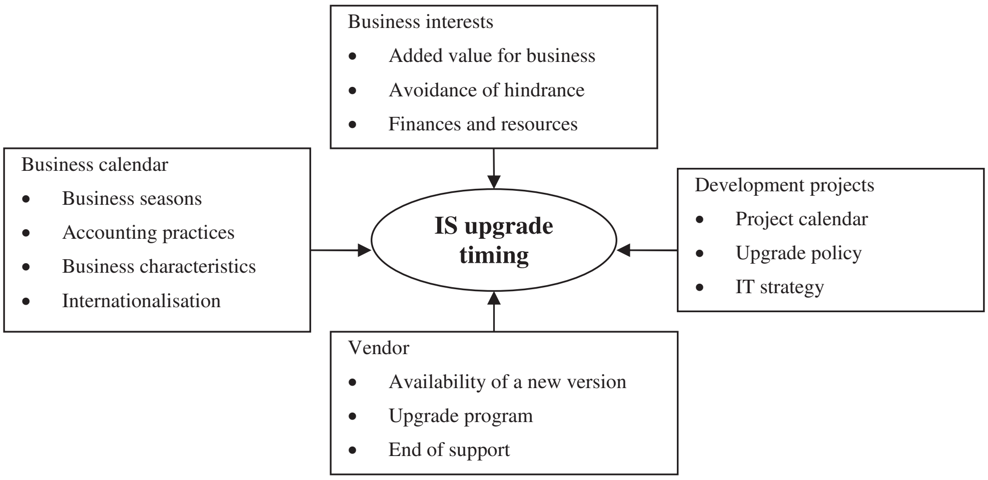

Figure 1. The determinants of IS upgrade timing.

The literature suggests that the possibilities for system upgrade for a customer are largely dependent on the phase of the product's life cycle. Our findings parallel. When the product's lifecycle is in its early phases, the vendor tries to persuade the customers for frequent version changes (see also Rajlich & Bennett 2000). Yet that is not necessarily the customer's interest (see also Esteves & Pastor 2005, Mukherji et al. 2006). During the servicing phase new version are no longer developed and in phase-out also the provision of support is terminated. Hence, the customer needs to understand the difference between software product's evolution and servicing (Rajlich & Bennett 2000), i.e. the software product life cycle. Our findings suggest that the interests of the customer and vendor often collide for the aforementioned reasons regarding the product life cycle. In Case K, conflict was a result from the customer's desire to continue the use of an aged version that the vendor was not able to support. However, the most commonly experienced conflict is due to the fact that a vendor persists on a version upgrade against the customer's interests.

Our findings indicate that the reasons that limit the possibilities and time-slots for system upgrade in the user organization are business calendar, including business seasons, accounting practices, business characteristics and degree of internationalization. Besides, business interests such as avoiding hindrance to the business and availability of resources and finances set boundaries and limitations for the timing of IS upgrading. Nevertheless, the most significant determinant for upgrade timing is the business value of the upgrade. An upgrade can be justified only if a profitable business case can be presented.

The technology upgrade model by Mukherji et al. (2006) states that the best time for an upgrade is when the gap between the version in use and new available version reaches a critical threshold. Our findings do not explicitly reveal whether the companies use such models for determining the suitable upgrade timing. However, the findings support "leapfrogging" as the most efficient and used technique for defining IS upgrade timing. Only in rare cases (e.g. case B) the actual upgrade plan was followed and each new version deployed. The predominant technique was to balance between using the old version as long as possible and changing versions as rarely as possible. This means that the customers have in-depth knowledge about the functionalities of the systems in use and their own needs. On the other hand, they actively follow the development of the products on the markets in order to be able to compare the functionalities with the functionalities of their present version. The degree of leapfrogging is determined by the aforementioned business calendar related issues and business interests.

Additionally it is influenced by the organizational development projects. Project calendar, internal policies and plans, organization's IT strategy define a frequency of system upgrades (e.g. case C and H).

IS upgrades are planned and scheduled in conjunction with other development activities in order to avoid overlapping projects (case J) or to combine projects when joint benefits are anticipated (cases A and O). This was considered as an important issue particularly concerning the fluency of work of the end-users.

Our investigations provide evidence on the importance of correct timing of IS upgrades. Business reasons are the main motives for defining IS version change or upgrade timing. They are strongly related to risk management, expected business benefits, and avoiding hindrance to business. Yet they are not the only reasons. Technological advancement and wear have also their impacts on upgrades. Despite of the dependency on the vendor, the customer has quite a lot of freedom with respect to upgrade timing. The customer may postpone the upgrade up to the point when it has to be made as e.g. support services are stopped (e.g. case K). The availability of a suitable version and resources are the preconditions for IS upgrade. Business calendar, business issues and development projects on their half define when in are the possible times for upgrade.

In the past, IS literature (e.g. Avison & Fitzgerald 2003, Iivari et al. 2009, Hartwick & Barki 1994) has greatly emphasized the vendors' needs to understand the customer. Likewise, the customer may benefit from understanding their software product's vendor. In order to gain competitive advantage, the customer need to carefully plan their IS upgrade activities in respect to appropriate versions available. If they can anticipate the changes in the product's lifecycle, they can choose their next upgrade and its schedule more beneficially. Based on the literature analysis and empirical findings, it can be concluded that if the customer-vendor relationship is framed by open climate and mutual trust, the conflict of interest stemming from the economics of upgrading (i.e. eagerness to sell and reluctance to buy) can be minimized.

7 SUMMARY

In this paper we have studied the determinants for the timing of enterprise system (ES), particularly ERP system, upgrades from user organization's point of view. We aimed at recognizing conditions that define system upgrade timing. We have collected, by interviewing 14 CIOs, a list of determinants that influence the IS upgrade timing. The empirical findings indicate that customer's IS upgrade timing is defined by business interests, business calendar, ongoing and future development projects, and the vendor. The paper contributes to practice by illustrating determinants for the timing of system upgrades in user organizations. The paper contributes to research by increasing the general understanding of the rarely studied phenomenon: system upgrades. Both vendors and customers may benefit by gaining a deeper understanding of the dynamics of IS upgrade timing by comprehending each other's motives and interests related to it.

In this study, our focus has been on the customers. In order to understand the phenomenon holistically, also the vendors should be studied by empirical means. The study was limited geographically in Finland. Thus the organizational culture provides nationally colored findings and limits the ability to generalize the results. Although most of the studied organizations are in global business, the results might be different in another context. The findings indicate that there are differences in how timing is defined in different fields of industry. Yet, our sample is too small to make any generalizations about and beyond these differences. We call for further research to investigate this issue. Despite these limitations, we believe the results are valuable for researchers and practitioners in their future activities of scheduling the next upgrade.

ACKNOWLEDGMENT

We are grateful to the Finnish Foundation for Economic Education and the University of Jyväskylä for supporting this research.

References

Al-Mudimigh, A., M. Zairi and M. Al-Mashari (2001). ERP software implementation: an integrative framework. European Journal of Information Systems 10, 216–226.

Avison, D. and G. Fitzgerald, G (2003). Where now for development methodologies?. Communications of the ACM, 46(1), 78-82.

Chen, I.J. (2001). Planning for ERP systems: analysis and future trend. Business Process Management Journal, 7 (5), 374-386.

Cooper, R.B. and R. W. Zmud (1990). Information technology implementation research: A technology diffusion approach. Management Science, 36 (2), 17-31.

Esteves, J.M. and J.A. Pastor (1999). An ERP Life-cycle-based Research Agenda. In Proceedings of the First International workshop in Enterprise Management and Resource Planning: Methods, Tools and Architectures – EMRPS'99, Venice, Italy.

Esteves, J.M. and J.A. Pastor (2005). A Critical Success Factor's Relevance Model for SAP Implementation Projects. Managing Business with SAP: Planning Implementation and Evaluation. Lau L. (Ed.), Hersley, PA, USA, Idea Group Publishing.

Hartwick, J. and H: Barki (1994). Explaining the role of user participation in information system use, Management Science, 40(4), 440-465.

Hong, K. and Kim, Y. (2002). The critical success factors for ERP implementation: an organizational fit perspective, Information and Management, 40 (1), 25-40.

Iivari, J., H. Isomäki and S. Pekkola (2009). The user - the great unknown of systems development: reasons, forms, challenges, experiences and intellectual contributions of user involvement. To appear: Information Systems Journal.

Kankaanpää, I. and P. Maaranen (2009). Considerations on ERP system upgrade timing in the light of vendor's and customer's interests. In the Proceedings of the 10th Conference of Association Information and Management (AIM).

Kankaanpää, I., P. Tiihonen, J.J. Ahonen, J. Koskinen, T. Tilus and H. Sivula (2007) Legacy System Evolution – A Comparative Study of Modernisation and Replacement Initiation Factors. In the Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS), DISI, INSTICC, 280-287.

Karsten, H. (1999). Collaboration and Collaborative Information Technologies: A review of the evidence. The DATA BASE for Advances in Information Systems, 30 (2), 44-65.

Markus, M.L. and C. Tanis (2000). The Enterprise System Experience — From Adoption to Success. Framing the domains of IT management: Projecting the Future Through the Past. R.W. Zmud and M.F. Price (Eds.), Pin-naflex Educational Resources, Inc.

Mehra, A. and A. Seidmann (2006). The Economics of Software Upgrades Throughout the Product Life Cycle. In the Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS 2006), IEEE Computer Society, Washington, DC.

Messerschmitt, D.G. and C. Szyperski (2003). Software Ecosystem. The MIT Press.

Mukherji, N., B. Rajagopalan and M. Tanniru (2006). A decision support model for optimal timing of investments in information technology upgrades. Decision Support Systems, 42, 1684-1696.

Narasimhan, R., S. Talluri and R.K. Mahapatra (2006). Multiproduct, Multicriteria Model for Supplier Selection with Product Life-Cycle Considerations. Decision Sciences, 27 (4), 577-603.

Ng, C.S.P. (2001). A Framework for Enterprise Resource Planning Maintenance and Upgrade Decisions. In the Proceedings of the 7th American Conference on Information Systems (AMCIS).

Ng, C.S.P. and G. Gable and T. Chan (2003). A Revelatory Case Study into the Adequacy of Standard Maintenance Models in an ERP Context. In the Proceedings of the 7th Pacific Asia Conference on Information Systems (PACIS), Adelaide, South Australia.

Rajlich, T.V. and K.H. Bennett (2000). A Staged Model for the Software Life Cycle. Computer 33 (7), 66-71.

Sawyer, S. (2001). A Market-based Perspective on Information Systems Development. Communications of the ACM, 44 (11), 97-102.

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

Stefanou, C. (2001). A Framework for the ex-ante evaluation of ERP software. European Journal of Information Systems, 10, 204-215.

Wagner, E.L. and S. Newell (2007). Exploring the Importance of Participation in the Post-implementation Period of an ES Project: A Neglected Area. Journal of the Association for Information Systems, 8 (10), 480-479.

Wei, X., C. Weiss and B.R. Nault (2007). Experience Information Goods: Versioning and Upgrading. In the Proceedings of the 2007 INFORMS Conference on Information Systems and Technology, Seattle, Washington.

Verville, J., C. Bernadas and A. Halingten (2005). So you're thinking of buying an ERP? Ten critical factors for successful acquisitions. Journal of Enterprise Information Management, 18 (6).

Appendix A: The interview instrument

Background information

(A recently upgraded IS was chosen for the topic of the interview.)

Name of the information system: ____

Purpose of the information system: ____

Vendor: ____

Year of acquisition:____

Interview questions

What kind of phases the IS has gone through during its life until the present date?

What reasons triggered the latest upgrade?

How did you get information of the available new versions?

On what grounds the upgraded version was selected?

On what grounds was the timing of the upgrade decided upon?

What kind of significance the timing has on IS upgrade?

How well the timing of this recently upgraded IS succeeded? ______ (On scale 1…..5, where 1 = poorly and 5 = outstandingly)

V

IS EVOLUTION BENEFIT ASSESSMENT – CHALLENGES WITH ECONOMIC INVESTMENT GRITERIA

by

Irja Kankaanpää, Jussi Koskinen, Tero Tilus, Henna Sivula, Heikki Lintinen & Jarmo J. Ahonen

Technologies for Business Information Systems (pp. 183-191)

IS Evolution Benefit Assessment – Challenges with Economic Investment Criteria

Irja Kankaanpää[1], Jussi Koskinen[1], Tero Tilus[1], Henna Sivula[1], Heikki Lintinen[1], Jarmo J. Ahonen[2]

[1]Information Technology Research Institute, University of Jyväskylä, P.O. Box 35, 40014 Jyväskylä, Finland
[2]Department of Computer Science, University of Kuopio, P.O. Box 1627, 70211 Kuopio, Finland

Abstract

In this paper, eight financial investment criteria and their advantages and disadvantages with respect to IS evolution benefit assessment are studied. As a result, it can be suggested that it is unattainable to define one appropriate evaluation method for IS evolution assessment in general. However, NPV and ROI appear to be the most appropriate methods for IS evolution evaluation. Their results can be reinforced with the supporting investment criteria, including AAR, IRR and payback method. The following potential challenges in evolution investment decision making were identified: selection of an appropriate criteria, acquisition of suitable metrics and follow-up data, data conversion, and comparison of different types of evolution options. A preliminary framework, ISEBA, was developed to address these challenges.

1 Introduction

Maintenance and system evolution activities have a significant role in the information system (IS) life cycle. It has been estimated that approximately 80% of the total IT expenses are allocated for maintenance activities [20]. According to Lehman's first law, maintenance is necessary, because software needs to be continuously improved or it will get out of date and cannot respond to the requirements of its environment [19]. Despite the importance of IS evolution investments, there is a gap between the IT related costs and company profitability [27]. Brynjolfssen [10] described this as a productivity paradox: information technology utilization has increased since the 70's but simultaneously productivity has slowed down.

The work effort of maintenance is generally proportional to the life time of a system. Therefore, it is more dominant in legacy information systems [7]. Besides being old, a legacy system is typically large at size and contains vital information for the user organization, uses out-of-date technology, and is laborious to maintain [3] [4]. There are three strategies to deal with a legacy system: 1) maintaining and using the system as it is, 2) developing or purchasing a new system to replace it, or 3) radically improve, i.e. modernize, the legacy system in order to meet the new business needs [4] [25, p 8-10].

IT investments can be roughly classified in two categories, acquisition projects and development projects [24]. In the context of IS evolution, acquisition project includes purchasing off-the-self software in order to replace the existing legacy system. Development investment refers to a project that aims at developing new or modernizing the existing system. Therefore, replacement can be either an acquisition or development project, while modernization is always a result of development activity. The major difference between acquisition and development projects, in terms of investment evaluation, is the length of time that is required for the benefits to start to appear [24]. In the first, an organization starts to benefit from the investment as soon as the acquisition has been made. In the latter, the benefits start to accumulate only after the project has been completed [24].

Evolution investments are economically significant and, consequently, their justification in financial terms is important. Because a legacy system is closely tied to an organization, a careful consideration of operational environment and organizational context is a prerequisite for its successful migration [5], and should be incorporated in the evaluation process. However, in reality the management often expects plainly financial evidence to support evolution decisions. In this paper, the goal is to study the advantages and disadvantages of financial investment criteria and their suitability in IS evolution benefit assessment. Additionally, a framework for evaluation method selection is presented.

This is a work-in-progress paper that summarizes the preliminary work on IS evolution benefit assessment within an industry co-operation project called ELTIS (Extending the Lifetime of Information Systems) during 2003-2005. The project was carried out in the Information Technology Research Institute (ITRI), University of Jyväskylä, Finland. It focused on prolonging the lifetime of IS in an economically viable manner.

2 Investment criteria

Dehning and Richardson [12] conducted a literature review on studies covering the impacts of IT on firms' performance in 1997-2001. In most of these studies, IT investments had been evaluated with the means of direct performance or accounting measures. That is where the business owner, by tradition, is expecting to see the implications of investments. In case of IS evolution investments, however, the benefits are not necessarily reflected on the firm's performance or accounting figures. The financial investment criteria can only detect tangible benefits while ignoring the intangibles. In the past, several benefit assessment methods have been developed in order to address this problem [15]. In this paper, the focus is on the so called classical financial investment evaluation methods. On the basis of a literature review, the advantages and disadvantages of eight financial investment criteria are presented and their suitability on IS evolution assessment is evaluated.

Classical financial investment criteria can be divided in three categories: 1) discounted cash flow criteria, 2) payback criteria, and 3) accounting criteria. Discounted cash flow criteria include *net present value* (NPV), *internal rate of return* (IRR), and *profitability index*. Payback criteria consist of *payback period* and *discounted payback* period. Accounting criteria consists of *average accounting return* (AAR). [23, p 256]. Other investment criteria include *return on investment* (ROI) method and *real options* approach (option pricing models) [11, p 139]. Investment criteria, their advantages and disadvantages are described in Table 1.

In general, discounted cash flow criteria are considered the most preferred option when evaluating investment proposals [23, p 256]. NPV is in most cases the recommended approach [23, p 256]. On the contrary, investment criteria based on accounting figures are not as useful with respect to investment planning is because they are aggregated and past-oriented [23, p 245]. They can be used, however, as complementary criteria together with other investment criteria.

Table 1. Investment criteria

Criteria and definition	Advantages	Disadvantages
NPV The difference between investment's market value and cost [23, p 233].	Includes time value of money [11, p 73]. No serious flaws [23, p 256].	Unsuitable for analyzing acquisitions because of short-term and user-oriented focus [13]. Unable to deal with uncertainty [24].
IRR The discount rate that makes the NPV of an investment zero [23, p 245].	Includes time value of money [11, p 73]. Results are easy to communicate and understand [23, p 253].	May lead to incorrect decisions if project cash flows are unpredictable, investment options are mutually exclusive, [23, p 253] or level of uncertainty is high [24].
Profitability Index The present value of an investment's future cash flows divided by its initial cost [23, p 253].	Results are easy to communicate and understand. Useful if investment funds are scarce. [23, p 253-254].	May give misleading results when investments options are mutually exclusive [23, p 254].
Payback period A time period from the moment when an investment is made to the moment when the cash flow from the investment equals the original investment cost [23, p 240].	Simple and easy to under-stand. Adjusts for uncertainty of later cash flows. [23, p 240]	Requires an arbitrary cut-off point. Ignores time value of money and cash flows beyond cut-off date. Biased against long-term or new projects, and liquidity. [23, p 240]
Discounted payback period The length of time required for an investment's discounted cash flows to equal its initial cost [23, p 240].	Includes time value of money [23, p 242].	Ignores cash flows beyond cut-off date [23, p 256]. Biased towards liquidity [23, p 242].
AAR An investment's average net income divided by its average book value [23, p 243].	Easy to calculate. Needed information is often available. [23, p 245]	Ignores the opportunity cost [9] and time value of money [23, p 245]. Does not compare to real market returns [9].
ROI The ratio of net benefits plus the original investment divided by the initial investment [11, p 70].	One of the most significant calculation methods for evaluating managerial performance [6, p 207]. Simple and clear [11, p 70].	Ignores the scale of the investment and timing of cash flows. Not useful for planning. [6, p 207]. Insufficient if used alone [11, p 72-73].
Real options An approach used to evaluate alternative management strategies using traditional option pricing theory [2].	Able to deal with uncertainty [24]. Provides managerial flexibility [11, p 146] [24]. Includes timing and risk [11, p 142].	Complex to communicate. Input values are difficult to estimate. Reliance on assumptions. [16]

Traditionally, the financial analysis of IT acquisition projects has been conducted with NPV or discounted cash flow analysis [13]. During the last few decades, also options pricing models have been applied in IT evaluation [24]. ROI has been used to evaluate the benefits of software reuse [20]. However, there are no reports on the use of these investment criteria in IS evolution decision making particularly.

In the context of IS evolution, the best suited financial evaluation methods are simple and require minimum use of resources. After comparing the characteristics of investment criteria, it can be concluded that discounted cash flow criteria and ROI would be appropriate considering those requirements. IRR and profitability index, however, may lead to incorrect results in the case of mutually exclusive investments, i.e. when accepting one investment prevents taking another [23, pp 253-254]. Therefore, they may not be suitable method for IS evolution options evaluation, since they in most cases are mutually exclusive. For instance, acquisition of a new system and modernization of the existing system most likely are investments from which only one is chosen. NPV has been criticized of its inability to deal with project uncertainty [24]. Because of that it may not be a preferred criterion in system modernisation evaluation. Real options approach seems to provide with the most holistic tool to compare (replacement) and development (modernization, replacement) projects. It can cope with the risks and uncertainty related to modernization and provides future oriented results. Also, it provides managerial flexibility allowing decisions about the investment to be changed as new information becomes available [11, p 146] [24]. However, it is mathematically demanding, which sets certain limitations to its use [16]. Also, it requires relatively detailed input data [16] which may cause the estimation method itself become heavy and uneconomical to use. Due to its past-oriented nature, accounting criterion is not useful with respect to IS evolution options evaluation.

In order to avoid one-dimensional view of an IT investment, use of more than one financial criteria is suggested [11, p 73] [23, p 254]. For instance, payback and AAR can be used to reinforce the results of NPV calculation [11, p 73]. The recommended methods for IT investment assessment, in general, are NPV, IRR and payback period [11, p 73]. Additionally, calculation should be conducted before and after the project [11, p 76].

3 Challenges

The main challenge in evaluating IT investments with financial criteria is the selection of a suitable benefit assessment method. As presented above, classical investment criteria are not uniformly suitable for every situation. If a method is selected carelessly, the results may recommend a refusal of a potential investment proposal simply because the selected method ignores a relevant factor [23]. Respectively, an unprofitable investment may seem potential if improper analysis methods are used.

The second challenge is related to existing and available data. In order to conduct a benefit assessment for investment options, a company has to gather IT-related data concerning its own activities to support management decision making [27]. This presumes the existence of a proper metrics program and follow-up. Without systematic data collection there is no accumulated history data on which the investment estimation could be based on. A related risk is that selected metrics do not capture the value of IT [27], i.e. insignificant or false metrics are being monitored.

The third challenge with economic criteria is that the benefits often appear in non-financial form [22, p 7] and the collected data is to be conversed in a commensurable format before benefit assessment can be carried out [14]. Data conversion may be problematic if benefits appear as soft issues, which are difficult to express in terms of money. In order to avoid confusion with data conversion, the expected benefits should be identified before data acquisition.

The fourth challenge, concerning particularly IS evolution evaluation, is in the comparison of different types of investment options. For instance, if all three evolution strategies are possible, there are minimum of three evolution options to be compared. Those can be for instance:

1. replace the existing system with system X (vendor X)
2. replace the existing system with system Y (in-house development)
3. modernize the existing system (in-house modernization)

Or, alternatively, the investment options can be:

1. replace the existing system with system X (vendor X)
2. replace the existing system with system Y (vendor Y)
3. modernize the existing system (vendor Y)

The successful comparison of investment option combinations necessitates that the selected evaluation method(s) are in compliance with the investment options and that the organization has the ability to use them accordingly.

As a conclusion, it can be suggested that it is unattainable to predefine the appropriate evaluation method for IS evolution options assessment.

Some of the investment criteria fit better than the others but in the end the selection should be made on the basis of the investment situation at hand combined with the available resources, skills, and data.

4 ISEBA framework

In the past research, various methods and approaches have been presented in order to merge IT evaluation with financial investment criteria. In 1987, Parker and Benson et al. [21] introduced information economics that seeks to unify financial justification, value, and innovation valuation with decision making. In 1992, Farbey et al. presented a model for matching an IT investment with a suitable evaluation method [15]. More resent models include a manager friendly roadmap for IT investment evaluation [28] and an evaluation method's matrix as a solution for customized IT investment evaluation [18]. A point of consensus for these methodologies is that their focus is on financial benefits of IT in general. Also, there are methods for software modernization cost and work effort estimation, i.e. COCOMO II [8], FPA [1] and Softcalc [26], but it seems that availability of methods for IS evolution evaluation is currently minimal. For this reason, a new framework for this particular purpose was outlined.

ISEBA (Information System Evolution Benefit Assessment) is a framework to support comparison of IS evolution options. Its goal is to provide assistance in the selection of a benefit evaluation method for investment situation at hand. It is based on empirical research consisting of interview study of industrial decision making and industrial co-operation projects, and a comprehensive literature survey. It obliges instructions about the required metrics and follow-up data for both intangible and financial evaluation methods. ISEBA consists of eight phases: 1) identifying the characteristics of investment situation, 2) identifying investment type, 3) defining investment assessment emphasis, 4) estimating organizational capabilities and comparing them to the requirements and labour intensity of potential benefit estimation methods, 5) selecting suitable method(s) and identifying related risks, 6) gathering required follow-up and metrics data for benefit assessment, 7) performing benefit assessment for investment proposals, and 8) interpreting and valuating results.

The implementation of ISEBA follows the form of a decision-tree. Phases 1 to 4 rule out the improper methods and provide a list of potentially suitable methods. In Phase 2, the investment type (acquisition or development project) of investment proposals is defined. This defines the post-investment measurement timing. The assessment emphasis in Phase 3 refers to financial or non-monetary benefits. It can be decided that either financial or intangible benefits or both are assessed depending on the investment characteristics. In phase 4, it is important to evaluate the resources and skills the organisation is able to allocate for evaluation. Also, the existing metrics and follow-up data are assessed. The final selection of suitable methods is based on the comparison on of the potentially suitable methods and organisation's resources, skills and available data (phase 5). ISEBA supports method selection by providing a description of each method and the required input data per method. In Phase 6, data acquisition (if needed) is carried out. The execution of benefit assessment takes place in phase 7. Finally, in phase 8 the results are examined and valuated in compliance with organization's strategies. If more than one investment proposal is to be evaluated, it is defined in Phase 1. A more detailed description of ISEBA is given in [17].

5 Summary

On the basis of the literature survey, it can be concluded, that the overall benefits of IS evolution have not been studied comprehensively so far. The comparison of the characteristics of investment criteria shows that the best suited criteria for IS evolution evaluation are NPV and ROI. IRR, payback, and AAR can be used as additional criteria in order to verify their results. Accounting criteria tend to be too general and past-oriented while real options rely strongly on assumptions and option prizing methods are demanding to apply. As a summary of the literature survey it can be concluded that there is no straightforward rule for defining the appropriate evaluation method for IS evolution options assessment. The selection should be made on the basis of the investment situation at hand combined with the available resources, skills, and data. Also, it is suggested that none of the investment criteria should be used alone but together with supporting criteria.

Potential challenges related to evolution investment assessment are selecting a proper analysis method, collecting suitable metrics data, data conversion, and comparison of different types of evolution options. Inspired by these challenges ISEBA framework was created. ISEBA has been further developed and validated empirically in real life cases which incorporate evolution benefit assessment. Report of the validated framework and completed two software industry related projects that

ELTIS promoted has been published [17]. Report of the further developed framework, named ISEBA+, will be forthcoming.

References

1. Albrecht A, Gaffney J (1983) Software function, source lines of code, and development effort prediction: a software science validation. IEEE Transactions on Software Engineering 9 (6) pp 639-648
2. Alleman J, Hirofumi S, Rappoport P (2004) An Investment Criterion Incorporating Real Options. In: Real Options – Theory Meets Practice, The 8th Annual International Conference, www.realoptions.org (on-line; read 23.8.2004)
3. Bennett K (1995) Legacy Systems: Coping with Stress. *IEEE Software* 12 (1), pp 19-23
4. Bennett K, Ramage M, Munro M (1999) Decision Model for Legacy Systems. *IEE Proceedings – Software* 146 (3) pp 153-159
5. Bergey JK, Northrop LM, Smith DB (1997) *Enterprise Framework for the Disciplined Evolution of Legacy Systems*. Technical report, CMU/SEI-97
6. Bierman H Jr, Drebin AR (1968) *Managerial Accounting: An Introduction.* The Macmillan Company, New York
7. Bisbal J, Lawless D, Wu B, Grimson J (1999) Legacy Information Systems: Issues and Directions. *IEEE Software* 16 (5) pp 103-111
8. Boehm B, Horowitz E, Madachy R, Reifer R, Clark B, Steece B, Brown A, Chulani S, Abts C (2000) Software Cost Estimation with COCOMO II. Prentice Hall
9. Brealey R, Myers SC (1996) Principles of Corporate Finance. 5th edn, McGraw-Hill, USA
10. Brynjolfssen E (1993) The Productivity Paradox of Information Technology. Communications of the ACM 36 (12) pp 67-77
11. Curley C (2004) Managing Information Technology for Business Value. Intel Press, Hillsboro, OR
12. Dehning B, Richardson VJ (2002) Returns on Investment in Information Technology: A Research Synthesis. Journal of Information Systems 16 (1) pp 7-30
13. Dos Santos BL (1991) Justifying Investments in New Information Technologies. Journal of Management Information Systems 7 (4) pp 71-89
14. Erdogmus H, Favaro J, Strigel W (2004) Return on Investment. IEEE Software 21 (3) pp 18-22
15. Farbey B, Land F, Targett D (1992) Evaluating Investments in IT. Journal of Information Technology 7 pp 109-122
16. de Jong B, Ribbers PMA, van der Zee HTM (1999) Option pricing for IT valuation: a dead end. EJISE Electronic Journal of Information Systems Evaluation, http://www.ejise.com/volume-2/volume2-issue1/issue1-art1.htm

17. Kankaanpää I, Sivula H, Ahonen JJ, Tilus T, Koskinen J, Juutilainen P (2005) ISEBA - A Framework for IS Evolution Benefit Assessment. In: Remenyi D (ed) Proceedings of the 12th European Conference on Information Technology Evaluation, Academic Conferences Ltd, UK, pp 255-264
18. Lech P (2005) Evaluation methods' matrix – A Tool for customized IT investment evaluation. In: Remenyi D (ed) Proceedings of the 12th European Conference on Information Technology Evaluation, Academic Conferences Ltd, UK, pp 297-306
19. Lehman MM, Perry DE, Ramil JF (1998) Implications of Evolution Metrics on Software Maintenance. In: Khoshgoftaar TM, Bennett K (eds) Proceedings of the International Conference on Software Maintenance, IEEE Computer Society, Los Alamitos, California, pp 208-217
20. Lim WC (1996) Reuse Economics: A Comparison of Seventeen Models and Direction for Future Research. In Proceedings of the 4th International Conference on Software Reuse, IEEE Computer Society, pp 41-50
21. Parker MM, Benson RJ, Trainor HE (1988) Information Economics – Linking Business Performance to Information Technology. Prentice-Hall, Singapore
22. Remenyi D, Money A, Sherwood-Smith M with Irani Z (2000) The Effective Measurement and Management of IT Costs and Benefits. 2nd edn, Butterworth-Heinemann, Oxford
23. Ross SA, Westerfield RW, Jordan BD (1998) Fundamentals of Corporate Finance. 4th edn, McGraw-Hill, USA
24. Schwartz ES, Zozoya-Gorostiza C (2000) Valuation of Information Technology Investments as Real Options. In: The 4th Annual Real Options Conference, www.realoptions.org (on-line, read 23.8.2004)
25. Seacord RC, Plakosh D, Lewis GA (2003) Modernizing Legacy Systems – Software Technologies, Engineering Processes, and Business Practices. Addison-Wesley, Boston, MA
26. Sneed HM (1995) Estimating the Costs of Software Maintenance Tasks. In: Proceedings of the 11th International Conference on Software Maintenance, IEEE Computer Society, pp 168-181
27. Verhoef C (2002) Quantitative IT Portfolio Management. Science of Computer Programming 45 pp 1-96
28. Videira A, da Cunha PR (2005) Evaluating IT Investments: A Managerial Friendly Roadmap. In: Remenyi D (ed) Proceedings of the 12th European Conference on Information Technology Evaluation, Academic Conferences Ltd, UK, pp 501-510

VI

ISEBA – A FRAMEWORK FOR IS EVOLUTION BENEFIT ASSESSMENT

by

Irja Kankaanpää, Henna Sivula, Jarmo J. Ahonen, Tero Tilus, Jussi Koskinen & Päivi Juutilainen

The Proceedings of the 12th European Conference on Information Technology Evaluation (ECITE 2005) (pp. 255-264)

ISEBA – A Framework for IS Evolution Benefit Assessment

Irja Kankaanpää[1], Henna Sivula[1], Jarmo J. Ahonen[2], Tero Tilus[1], Jussi Koskinen[1], Päivi Juutilainen[1]
[1]University of Jyväskylä, Information Technology Research Institute, Jyväskylä, Finland
[2]University of Kuopio, Department of Computer Science, Kuopio, Finland
irja.kankaanpaa@titu.jyu.fi
henna.sivula@titu.jyu.fi
jarmo.ahonen@uku.fi
tero.tilus@titu.jyu.fi
jussi.koskinen@titu.jyu.fi
paivi.juutilainen@titu.jyu.fi

Abstract: Decisions regarding information system evolution strategy, including modernization or replacement, are economically significant. Selection of a proper method for analyzing potential options, acquisition of suitable metrics or follow-up data, and evaluation of the results are major challenges in the evolution strategy decision making. In order to address these challenges, a framework for Information System Evolution Benefit Assessment (ISEBA) was developed. ISEBA provides assistance in the selection of a benefit-evaluation method for the investment situation at hand. It is based on empirical research on industrial decision making and co-operation projects, and examination of existing research results from information technology investment-evaluations. Based on the experiences from real world industrial cases, it can be alleged that ISEBA suits its purpose. It makes a decision maker take note of relevant issues in evolution investment decision making, leads to the selection of suitable appraisal method, and promotes discussion of the value of the results. Among the results of the industrial cases, the following was found. Firstly, resources are an important factor influencing the selection of an investment assessment method. Secondly, the appearance of both tangible and intangible benefits is scattered with respect to time and organizational structure. Thirdly, a dichotomy exists between upper management decisions and lower level organizational goals. The benefit appraisal was focused on an operative level while real project objectives were set on a strategic level. It is concluded, that a need for a practical framework, such as ISEBA, exists.

Keywords: IS evolution strategy, benefit assessment, industrial decision making, investment options evaluation

1. Introduction

Maintenance has a significant role in the information system (IS) life cycle. Depending on the publication time and scope of studies, 50-90% of the total IS expenses are caused by operation and maintenance activities (Lientz and Swanson, 1981; Erlikh, 2000). According to Lehman's first law, maintenance is necessary because software needs to be continuously improved or its ability to respond to the requirements of its environment decreases in time (Lehman et al., 1998). The inability to modify software rapidly may cause difficulties in exploiting new market opportunities (Bennett, 1995).

The amount of resources used on system evolution activities increases as information systems age. This creates the so called "legacy dilemma" (Bennett, 1995). On one hand, a legacy system is often business critical and has accumulated vital information during time. On the other hand, it is typically large in size, uses out-of-date technology and is laborious to maintain. Yet, it remains in operation because user organization cannot manage without it. There are three elementary strategies (Bennett, 1995; Seacord et al., 2003: 8-10) to deal with a legacy system: 1) continuing maintenance, 2) replacing the existing system, or 3) modernizing it. Modernization means improving the system in such a radical manner that it cannot be classified as traditional maintenance while a significant proportion of the legacy system is conserved (Seacord et al., 2003: 9). Replacement implies that existing legacy system is deserted and a substituting system is either purchased or developed (Seacord, 2003: 10).

Modernization and replacement are significant economical investments. Consequently, decisions regarding these should be made carefully. IT investment options can be analyzed with respect to their value, cost, and risks (Parker et al., 1988: 5) or the benefits that they yield (Ashurst and Doherty, 2003). In this paper, the term benefit is defined as the *overall benefit* of system evolution investment, i.e. the sum of positive and negative impacts, which can be either tangible or intangible or both.

In a literature survey, two frameworks that support the selection of benefit evaluation methods were found (see Farbey et al., 1992; Andresen 2002). In this paper, the existing models are re-developed and fitted into the IS evolution context. As a result, a model for IS Evolution Benefit Assessment (ISEBA), is presented. The rest of this paper is organized as follows. In chapter 2, current state of IT investment evaluation is discussed. Chapter 3 presents the ISEBA framework. Chapter 4 reports on the experiences when applying ISEBA in real world IS evolution cases. Finally, a summary and conclusions are given in chapter 5.

2. Evaluation of IT Evolution Investments

Recent research results show that there is an increasing pressure to show return on IT investments but IT executives and decision makers feel unable to compute it accurately (Verhoef, 2002; Koskinen et al., 2005). This is not due to a lack of evaluation methods but rather due to their oversupply. Andresen (2002) found in his literature survey 82 IT investment evaluation methods, each of which is suitable for specific type of investment situation. These include methods for measuring both tangible (i.e. quantitative or financial) and intangible (i.e. qualitative) benefits. The selection of a proper method from such a variety can be seen as a major challenge. Andresen's list of evaluation methods was delimited and supplemented with the methods that were recognized useful in real life IS evolution evaluation cases described in chapter 4 (see Table 1).

Table 1: IS evolution evaluation methods (Adapted from Andresen 2002).

Tangible methods	Intangible Methods
Accounting rate of return (ARR)	Critical success factors (CSF)
Capital investment appraisal	Delphi approach
Cost estimation	Exit of problems
Cost-benefit analysis (CBA)	Expectations analysis
Cost-effect analysis (CCA)	Experimental methods
Cost-revenue analysis (CRA)	Multi objective – multi criteria (MOMC)
Critical success factors	Portfolio analysis
Discounted cash flow analysis (DCF)	Present state analysis
Discounted payback method	Favored functionality analysis
Information economics (IE)	User satisfaction survey
Internal rate of return (IRR)	Value analysis
Net present value (NPV)	
Payback method	
Profitability index (PI)	
Real options method	
Return on investment (ROI)	
Return on management (ROM)	
Savings in employee-work-years	
Total cost of ownership (TCO)	

Dehning and Richardson (2002) conducted a literature review on studies covering the impacts of IT on firms' performance in 1997-2001. In most of these studies, IT investments have been evaluated with the means of direct performance measures or accounting measures. Financial investment evaluation methods, however, are not uniformly suitable or sufficient (if used alone) for every situation. If a method is selected carelessly, the results may recommend a refusal of a potential investment proposal simply because the selected method ignores a relevant factor (Ross et al., 1998). Respectively, an unprofitable investment may appear attractive if improper analysis methods are used. The fundamental ideology that influences the judgment of IT investment options is that the benefits should be greater than the costs. This leads the decision makers quite naturally to a situation where they try their best to measure the benefits in terms of money even if it does not seem quite possible. In the context of IS evolution, the impact of modernization or replacement may appear in form of intangible benefits, which cannot be detected by financial measures. In those cases, value can be used as a substitute of benefit (Parker et al., 1988: 64). On the other hand, modifying the intangible benefits into monetary terms may be so resource consuming that evaluation itself becomes unprofitable.

Besides the method selection, the acquisition of suitable metrics or follow-up data and evaluation of the results are major challenges in the evolution strategy decision making. In order to conduct a benefit assessment for investment options, a company needs to gather IT-related data concerning its own activities to support management decision making (Verhoef, 2002). This presumes the existence of a proper metrics program and follow-up. A related risk is that selected metrics do not capture the value of IT (Verhoef, 2002). Metrics and follow-up data are to be conversed in a commensurable format before a cost-benefit or other analysis can be carried out (Erdogmus et al., 2004). Data conversion may be tricky if benefits appear as soft issues, which are difficult to express in terms of money. In order to avoid confusion with data conversion, the expected benefits should be identified before data acquisition.

A careful consideration of operational environment and organizational context is a prerequisite for a successful IS migration (Bergey et al., 1997). Issues to be considered are: role of the evaluation, decision environment, role of the system, characteristics of the organization, and cause and effect relationships (Farbey et al., 1992).

As a solution to the first challenge, Farbey et al. (1992) developed a model for matching an IT project with an evaluation method. A suitable method for a project is defined by locating the situational factors into a matrix and comparing the result to the "techniques matrix". A decade later, Andresen (2002) continued Farbey's work and presented an extended framework on how to select an IT evaluation method in the context of construction industry. What is missing in Andresen's model, however, is transparency regarding the suggestion of a suitable evaluation method. Also, Andresen's model gives one an impression that after method selection, most of the work is done. In real life though, application of the selected method and executing related tasks are the most labour intensive activities in benefit assessment. A method may suit the evaluation purpose perfectly but the needed input data is not available or its acquisition requires work hours beyond limits. Additionally, Andresen's model does not provide guidance or instructions about the required data as an input for the suggested method. It does not take into consideration unsuitable methods but instead focuses only on finding suitable ones.

An interview study of 29 decision makers representing software supplier and user organizations showed that the problems in decision making are mostly related to a non-systematic and narrow-viewed approach. The interviewees were, in general, unsatisfied with their informal decision-making process and thought that estimates of benefits were difficult to achieve. They were hoping to be provided with tools to systematize the process and means to widen the scope of thinking and to acquire a new perspective. In this way, fabricated arguments for or against evolution options could be minimized. Also, information about the previous decisions was considered important if it was available. A prerequisite for a support method is that it is transparent in order to enhance the thinking process. A "black box" method is of no use. The complete results of the interview study are reported by Koskinen et al. (2005).

3. Framework for IS evolution benefit assessment

ISEBA is a framework for IS evolution benefit assessment. It is based on empirical research consisting of an interview study of industrial co-operation projects, and examination of the challenges of benefit estimation for IS evolution options. ISEBA provides assistance in selecting suitable, and ruling out improper methods, for an investment situation. Also, it obliges instructions about the metrics and follow-up data that the selected method requires as input. ISEBA consists of four phases: 1) Determination of method applicability, 2) Selection of methods, 3) Implementation of the benefit assessment, and 4) Decision making (see Figure 1).

Phase 1 consists of the following activities: Evaluation of investment characteristics and environment, identification of investment type (Schwartz and Zozoya-Gorostiza, 2000) and defining investment assessment emphasis, i.e. evaluation criteria. As a result of the activities in phase 1, the unsuitable methods for selected criteria are excluded from the pool of potential methods. A method is classified unsuitable if there is a risk that it produces misleading results on the investment proposal in the defined context or is unable to measure selected criteria. If there is a method that measures the selected criteria better than others, then this method gets a "mark of recommendation", which increases the odds of remaining in the pool of potentially suitable methods. A method is classified suitable if evidence of the suitability exists or if the method has no clear reason to be unsuitable. The classification of "suitable" or "unsuitable" is based on literature and real life experiences using ISEBA.

In phase 2, the selection of suitable benefit assessment method(s) takes place. The amount of recommendations provides reference of the suitable methods per evaluation criteria. The excluded criteria cannot appear in the group of recommended methods. Additionally, instructions on required metrics and follow-up data for recommended methods are given. In Phase 3, data acquisition is carried out and benefit assessment of selected evolution options is performed. Use of a systematic measurement planning method, such as GQM (Basili, 1992), is recommended. Finally, in phase 4 the results of benefit assessment are analyzed and valuated in compliance with organization's strategies. A decision is made regarding the most fit option. If more than one investment proposal is to be evaluated, it is defined in Phase 1. It is assumed, that prior to use of ISEBA, the potential evolution options have been identified. If there is uncertainty about the possible evolution options, use of a scenario tool, such as SABA (Bennett et al., 1999), is recommended.

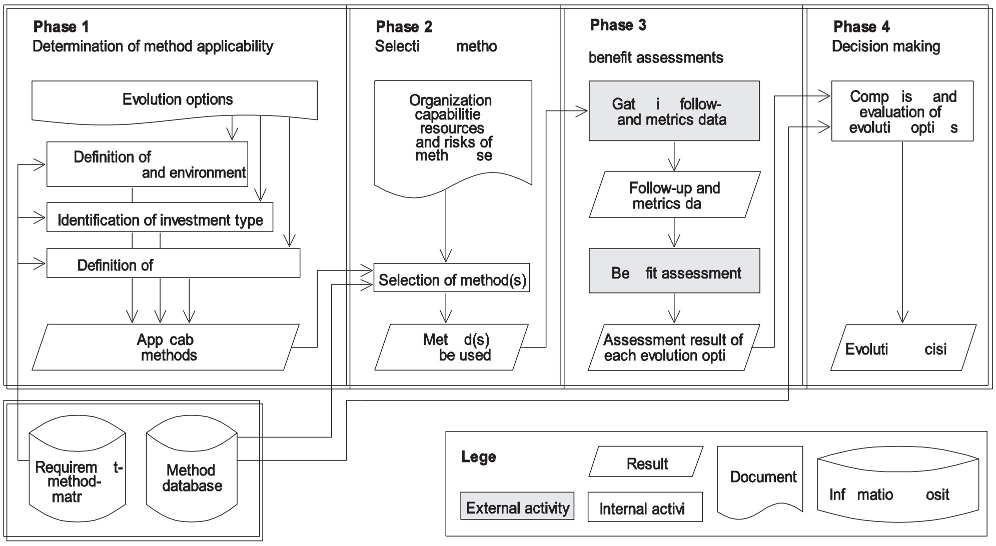

Figure 1: Structure of ISEBA.

ISEBA includes a method database consisting of the following information to support the framework: a requirement-method-matrix, description of required inputs, resources, and skills per method, and general guidelines for the interpretation of the results. As a result of phase 1, evaluation criteria form a "requirements profile" against which the suitability of methods is reflected. The requirement-method-matrix provides a listing of suggested (S) and not recommended (N) methods per evaluation criteria. Presently, it consists of 13 tangible evaluation criteria, 8 intangible evaluation criteria, and 19 criteria concerning the evaluation method, which can be related to 29 evaluation methods. The list of criteria and methods is up-dated on a regular basis as the research proceeds. An example of the requirement-method matrix is described in Table 2.

Table 2: An example of requirement-method-matrix.

Criteria	Attributes	AAR	CBA	IE	ROI	Value analysis
Use of resources	Evaluation should not use much resources. Inexpensive.	S	N			N *(Farbey et al., 1993)*
	Organization is prepared to allocate resources on the evaluation. Fairly expensive.			S *(Farbey et al., 1993)*	S *(Farbey et al., 1993)*	
	Carried out by professionals, involves various stakeholders or senior management. High cost.		S *(Farbey et al., 1993)*			S *(Farbey et al., 1993)*
Complexity of evaluation method	Simple and easy to use	S *(Ross et al., 1998: 245)*		N	S *(Curley, 2004: 70)*	
	May require professionals skills				S *(Farbey et al., 1993)*	
	Evaluation method can be complex			S *(Farbey et al., 1993)*		

A description of required inputs, resources, and skills per method, and general guidelines for the interpretation of the results are similarly parameters that are maintained as more methods and information become available.

The structure of ISEBA is described in more detail in Table 3. Description, prerequisites, tasks, and results are provided for each activity depicted in Figure 1.

Table 3: ISEBA, detailed description.

Phase	Activity	Description	Prerequisites	Tasks	Results
Phase 1	Evaluation of investment characteristics and environment	Investment characteristics and environment are defined.	Evolution options have been identified.	Consider • pressure for evolution investment • related uncertainty • relationship between evolution options	A preliminary elimination of unsuitable methods.
	Identification of investment type	Defines the post-investment evaluation timing.	Evolution options have been identified and agreed upon.	Identify the investment type • development project • purchase project	A preliminary estimate of the post-implementation benefit assessment timing.
	Definition of evaluation criteria	Evaluation criteria is defined and valuated with respect to the project goals.	Evolution options have been identified and agreed upon.	Consider • Which factors in an organization are most likely to reflect the potential benefits? • Are the potential benefits tangible or intangible? • On what business level will the benefits appear (operational or strategic level)? • Which criteria are more important than others?	Requirements profile. A list of suitable and not suitable evaluation criteria.
Phase 2	Selection of method(s)	The selection of potential methods takes place against requirements profile defined in phase 1. Information about input requirements of potential methods is provided by method database.	Evaluation criteria have been agreed upon and selections of phase 1 have been accomplished. List of potential methods is formed and available.	• Risks related to the evaluation methods are acknowledged. • The required inputs are assessed. • Complexity of methods is evaluated and compared to organizational skills. • Resource consumption of methods is assessed and compared to the available resources.	A list of evaluation methods to be applied.
Phase 3	Gathering follow-up and metrics data	The required input data for selected method is collected and processed.	Evaluation method(s) have been agreed upon.	• Data collection is planned. • Data collection and carried out.	A collection of processed data in the required format.
	Benefit assessment	Benefit assessment is carried out in accordance with the selected method.	The required data is available in required format.	• Benefit assessment is carried out. • Results of benefit assessment are summarized.	A summary of the evolution options benefit assessment.
Phase 4	Comparison and evaluation of evolution options	Decision about the potential evolution options is made. Attention should be paid to the reliability, accuracy and timing of benefit assessment results, and political interests of different stakeholders.	Sufficient amount of information of the potential evolution options is available and in commensurable format to support decision making.	• Results are analyzed with respect to the evaluation criteria and project goals. • Options are compared.	Organization is able to select the most beneficial evolution option that suits the organizational situation.

4. Experiences with using ISEBA

In this chapter, experiences of two real world industrial cases are discussed. The phase of ISEBA is referred with phase number (1-4) in the beginning of corresponding columns. Three parties were involved in the use of ISEBA: the user organization, the system vendor and the independent, external research group. Background information is provided as much as possible within the limitations of confidentiality.

Case 1

In case 1, the user organization was in the process of decision making concerning the evolution strategy of their invoicing system that was developed in the late 80's. The usability and maintenance of the system caused problems in everyday work. The existing invoicing system was originally tailored to comply with the organization's complex and unique business process. The introduction of a corresponding commercial off-the-shelf product was considered but it was discarded due to difficulty and high cost. Replacement with off-the-shelf product would have caused serious changes to the business operations and development of a new replacing system would have been extremely expensive. Hence, the possible evolution options were modernization or continuing the maintenance of the existing system.

(1) The organization was particularly interested in the potential benefits of modernization. The goal was to evaluate the potential effects of modernization from business value perspective and to produce information to support decision making. The emphasis of the evaluation was on intangible benefits appearing within a short period of time (c. 2 years). User satisfaction and system's suitability to organization's business processes were identified as important valuation criteria. In order to make a justified decision, project introducer needed more information on these issues. The criteria were a result of a meeting where representatives of user organization, vendor and independent researchers were present.

(2) Use of company resources was a central selection criterion for benefit assessment methods. The organization had a limited amount of work hours allocated on the benefit estimation. However, the participation of external researchers enabled the use of resource consuming methods. On the basis of these requirements, the analysis of user satisfaction and problem situations was selected as benefit assessment methods.

(3) Data gathering took place in two intervals. At first, user satisfaction was charted by an internet-based questionnaire with 30 respondents. After that, 25 employees took part in a four week problem situation follow-up. Data processing and analysis, and reporting of the results formed a significant portion of the total work hours.

(4) The results of data gathering and analysis indicated the current level of user satisfaction and problems related to system use. The representatives of user organization were surprised by the results of the user satisfaction questionnaire. Contrary to expectations, most of the system users were relatively satisfied and happy to use the current system. When analyzing the results of the benefit assessment, the results were compared to the overall organizational goals and strategies. As a result of the evaluation of modernization option, the organization decided not to modernize the existing system at the time being. Instead, a renewal of user interface in the future was suggested by the researchers.

Case 2

A decision concerning the strategy of a customer register developed during the early 80's was agreed upon. This legacy system was a central tool in the organization's business and had nearly 200 users. Maintenance was difficult and the system did not meet with the organization's present data requirements. Also, it was restricting company's future development options. Replacement was considered extremely costly. Modernization seemed as a natural option because the organization had a well established, long-term relationship with the system supplier.

The purpose of the evaluation was to verify the benefits achieved by the modernization on operative level. In the long run, the aim was to develop the organization's decision making process and adopt new working practices. A secondary goal was to assure system users of the positive impacts of modernization and by doing so minimize possible resistance of change. Top management had a vision that modernization of their customer register would enable new business opportunities in the future, and even more importantly, would ensure company's survival in the rapidly changing markets. Measurement of these benefits, however, was considered difficult and resource consuming. Therefore, it was agreed that the benefits were not to be measured on a strategic level.

(1-2) Evaluation criteria were discussed in several meetings with the representatives of user organization, system supplier and independent researchers. As result, the technical quality and business value of the system, including user satisfaction, exit of system use related problems, preserving of system's good qualities, were selected as evaluation criteria. Tangible, i.e. financial benefits, were also considered interesting. It was soon realized, however, that data needed for financial calculations was only partially available and completing it afterwards would have required a significant amount of work. Expected benefits were only partially transformable into monetary units. Nevertheless, it was agreed that the available financial data should be utilized. On the basis of the requirement profile, it was concluded that present state and expectations analysis combined with a free form cost-effect analysis were the most suitable benefit assessment methods.

(3) For data collection, two measurements were planned, one before and one after modernization. In this way, the changes caused by the modernization could be detected. The focus of the evaluation was on system users. Data collection methods partially were chosen and developed on the basis of the previous industrial cases. As in case 1, user satisfaction was measured by a web-based questionnaire. System use related problems were charted by interviewing end users and following system use for a one week period. The favored qualities of the current system were studied by interviewing end users. The technical and business values of the customer register were discussed during interviews with technical and business experts representing both user and supplier organizations. After the completion of the project, the same measurements will be repeated. Approximately 100 employees took part in the data collection.

(3) Collected data complemented the previously made present state analysis by the user organization and vendor. A significant innovation was that in addition to problems and faults, also the good qualities of the current system should be charted. The identification of preferred features would enable the vendor to preserve them in the modernized system. Without this realization, there would have been a risk that those well functioning features could have been partially or completely replaced with less satisfying functionality.

(4) The results of current state and expectations analysis showed that the end users were somewhat more satisfied with the current system than was anticipated. During the problem follow-up period, fewer problems were reported than was expected. Technical evaluation verified that there were serious faults in the maintenance of the current system. The business value analysis showed that the customer register did not comply with the organizations current needs as it did a decade ago. These results were evaluated with respect to the measurement reliability and the organizational position of the persons who participated in data collection. The results of the current state and expectation analysis supported the decision to modernize.

5. Summary and conclusions

On the basis of the literature survey, it can be concluded, that the overall benefits of software evolution have not been studied comprehensively so far. Potential challenges related to evolution investment assessment are selecting an evaluation method, collecting required data, and valuation of the results. Inspired by these challenges ISEBA framework was created.

The experiences of case studies showed that ISEBA has the following advantages compared to an ad hoc decision process. Firstly, it makes a decision maker take note of relevant issues in evolution investment decision making. The structure of ISEBA incorporates different viewpoints towards evolution options. By following the framework the relevant issues are easily remembered. Secondly, the application of ISEBA leads to the selection of suitable appraisal method, and promotes discussion of the value of the results. In an early phase of system evolution decision making, the data requirements of potential methods are discussed. When doing so, it can be seen which input data is available or collectable. Through conversation this becomes clear to all stakeholders. Also, when selecting the data collection methods, the organization's maturity on their use should be considered. Familiar methods are more easily accepted than new ones. The time aspect is relevant to development projects where the first benefits may emerge only several months after the completion of the project. Therefore, timing of the benefit evaluation should be carefully planned.

Among the results of the industrial cases, it was found that resources are an important factor influencing the selection of an investment assessment method. Therefore, the labour intensity of potential methods should be known as early as possible. Also, the user organization should decide how much resources they can allocate on evolution options evaluation before they start the evaluation process. Additionally, it was found that, the benefits of modernization tend to appear in both tangible and intangible format and are scattered with respect to time and organizational structure. Because of this reason the viewpoints of different organizational levels (i.e. users, maintainers, managers) should be taken into consideration. For instance, the top management often sees the benefits from strategic perspective while middle-level management or end users may consider operational benefits more relevant. An interesting finding in case 1 was that, a dichotomy exits between upper management decisions and lower level organizational goals. The benefit appraisal was focused on an operative level while real project objectives were set on a strategic level. These different views should be compared to the goals of the intended evolution activities in order to find out the core benefits.

Based on the experiences from real life cases, it can be alleged that ISEBA suits its purpose. However, the results presented in this paper are derived from only two cases and therefore cannot be generalized as such. In the future, ISEBA as a framework requires more empirical validation. The method information database should be also be complemented.

6. References

Ashurst, C. and Doherty, N. F. (2003) "Towards the Formulation of a "Best Practice" Framework for Benefits Realisation in IT Projects", *Electronic Journal of Information Systems Evaluation*, Vol. 6, No. 2, pp. 1-10, Available: http://www.ejise.com/volume6-issue2/vol6-i2-articles.htm, [3.6.2005].

Andresen, J. L. (2002) "How to Select an IT Evaluation Method – In the Context of Construction", *Proceedings of CIB w78 Information Technology in Construction Conference*, Available: http://itc.scix.net/cgi-bin/works/Show?w78-2002-16, [20.5.2005].

Basili, V. R. (1992) *Software Modeling and Measurement: The Goal/Question/Metric Paradigm*, Technical Report CS-TR-2956, College Park, USA: Department of Computer Science, University of Maryland.

Bennett, K. (1995) "Legacy Systems: Coping with Success", *IEEE Software*, Vol. 12, No. 1, pp. 19-23.

Bennett, K. H., Ramage M. and Munro M. (1999) "Decision model for legacy systems", *IEE Proceedings - Software*, Vol. 146, No. 3, pp. 153-159.

Bergey, J. K., Northrop, L. M. and Smith, D. B. (1997) *Enterprise Framework for the Disciplined Evolution of Legacy Systems*, Technical report, CMU/SEI-97-TR-007, Pittsburgh: Software Engineering Institute, Carnegie Mellon University.

Curley, M. (2004) *Managing Information Technology for Business Value*, Hillsboro, OR, USA: Intel Press.

Dehning, B. and Richardson, V. J. (2002) "Returns on Investment in Information Technology: A Research Synthesis", *Journal of Information Systems*, Vol. 16, No. 1, pp. 7-30.

Erdogmus, H., Favaro J., and Strigel, W. (2004) "Return on Investment", *IEEE Software*, Vol. 21, No. 3, pp. 18-22.

Erlikh, L. (2000) "Leveraging Legacy System Dollars for E-Business", *IEEE IT Professional*, Vol. 2, No. 3, pp. 17-23.

Farbey B., Land F. and Targett D. (1992), "Evaluating Investments in IT", *Journal of Information Technology*, Vol. 7, pp. 109-122.

Farbey B., Land F. and Targett D. (1993), *How to Assess your IT Investment – A study of methods and practice*, Oxford, UK: Butterworth-Heinemann Ltd.

Koskinen, J., Lintinen H., Ahonen J. J., Sivula H., Tilus T. and Kankaanpää I. (2005) *Empirical Study of Industrial Decision Making for Software Modernizations*, Technical report, Jyväskylä, Finland: Information Technology Research Institute, University of Jyväskylä.

Lehman M. M., Perry D. E. and Ramil, J. F. (1998) “Implications of Evolution Metrics on Software Maintenance”, *Proceedings of the International Conference on Software Maintenance,* IEEE Computer Society, pp. 208-217.

Lientz, B. P. and Swanson, B. (1981) “Problems in Application Software Maintenance”, *Communications of the ACM*, Vol. 24, No. 11, pp. 763-769.

Parker, M. M., Benson, R. J., Trainor, H. E. (1988) *Information Economics – Linking Business Performance to Information Technology*, Singapore: Prentice-Hall.

Ross, S. A., Westerfield, R. W. and Jordan, B.D. (1998) *Fundamentals of Corporate Finance*, 4th edition, USA: McGraw-Hill.

Schwartz, E. S. and Zozoya-Gorostiza, C. (2000) “Valuation of Information Technology Investments as Real Options”, The 4th Annual Real Options Conference, Real Options Group, Available: http://www.realoptions.org/abstracts/abstracts00.html, [23.8.2004].

Seacord, R. C., Plakosh, D. and Lewis G. A. (2003) *Modernizing Legacy Systems – Software Technologies, Engineering Processes, and Business Practices*, Addison-Wesley.

Verhoef, C. (2002) “Quantitative IT Portfolio Management”, *Science of Computer Programming*, Vol. 45, pp. 1-96.

1 Ropponen, Janne, Software risk management - foundations, principles and empirical findings. 273 p. Yhteenveto 1 p. 1999.
2 Kuzmin, Dmitri, Numerical simulation of reactive bubbly flows. 110 p. Yhteenveto 1 p. 1999.
3 Karsten, Helena, Weaving tapestry: collaborative information technology and organisational change. 266 p. Yhteenveto 3 p. 2000.
4 Koskinen, Jussi, Automated transient hypertext support for software maintenance. 98 p. (250 p.) Yhteenveto 1 p. 2000.
5 Ristaniemi, Tapani, Synchronization and blind signal processing in CDMA systems. - Synkronointi ja sokea signaalinkäsittely CDMA järjestelmässä. 112 p. Yhteenveto 1 p. 2000.
6 Laitinen, Mika, Mathematical modelling of conductive-radiative heat transfer. 20 p. (108 p.) Yhteenveto 1 p. 2000.
7 Koskinen, Minna, Process metamodelling. Conceptual foundations and application. 213 p. Yhteenveto 1 p. 2000.
8 Smolianski, Anton, Numerical modeling of two-fluid interfacial flows. 109 p. Yhteenveto 1 p. 2001.
9 Nahar, Nazmun, Information technology supported technology transfer process. A multi-site case study of high-tech enterprises. 377 p. Yhteenveto 3 p. 2001.
10 Fomin, Vladislav V., The process of standard making. The case of cellular mobile telephony. - Standardin kehittämisen prosessi. Tapaustutkimus solukkoverkkoon perustuvasta matkapuhelintekniikasta. 107 p. (208 p.) Yhteenveto 1 p. 2001.
11 Päivärinta, Tero, A genre-based approach to developing electronic document management in the organization. 190 p. Yhteenveto 1 p. 2001.
12 Häkkinen, Erkki, Design, implementation and evaluation of neural data analysis environment. 229 p. Yhteenveto 1 p. 2001.
13 Hirvonen, Kullervo, Towards better employment using adaptive control of labour costs of an enterprise. 118 p. Yhteenveto 4 p. 2001.
14 Majava, Kirsi, Optimization-based techniques for image restoration. 27 p. (142 p.) Yhteenveto 1 p. 2001.
15 Saarinen, Kari, Near infra-red measurement based control system for thermo-mechanical refiners. 84 p. (186 p.) Yhteenveto 1 p. 2001.
16 Forsell, Marko, Improving component reuse in software development. 169 p. Yhteenveto 1 p. 2002.
17 Virtanen, Pauli, Neuro-fuzzy expert systems in financial and control engineering. 245 p. Yhteenveto 1 p. 2002.
18 Kovalainen, Mikko, Computer mediated organizational memory for process control. Moving CSCW research from an idea to a product. 57 p. (146 p.) Yhteenveto 4 p. 2002.
19 Hämäläinen, Timo, Broadband network quality of service and pricing. 140 p. Yhteenveto 1 p. 2002.
20 Martikainen, Janne, Efficient solvers for discretized elliptic vector-valued problems. 25 p. (109 p.) Yhteenveto 1 p. 2002.
21 Mursu, Anja, Information systems development in developing countries. Risk management and sustainability analysis in Nigerian software companies. 296 p. Yhteenveto 3 p. 2002.
22 Seleznyov, Alexandr, An anomaly intrusion detection system based on intelligent user recognition. 186 p. Yhteenveto 3 p. 2002.
23 Lensu, Anssi, Computationally intelligent methods for qualitative data analysis. 57 p. (180 p.) Yhteenveto 1 p. 2002.
24 Ryabov, Vladimir, Handling imperfect temporal relations. 75 p. (145 p.) Yhteenveto 2 p. 2002.
25 Tsymbal, Alexey, Dynamic integration of data mining methods in knowledge discovery systems. 69 p. (170 p.) Yhteenveto 2 p. 2002.
26 Akimov, Vladimir, Domain decomposition methods for the problems with boundary layers. 30 p. (84 p.). Yhteenveto 1 p. 2002.
27 Seyukova-Rivkind, Ludmila, Mathematical and numerical analysis of boundary value problems for fluid flow. 30 p. (126 p.) Yhteenveto 1 p. 2002.
28 Hämäläinen, Seppo, WCDMA Radio network performance. 235 p. Yhteenveto 2 p. 2003.
29 Pekkola, Samuli, Multiple media in group work. Emphasising individual users in distributed and real-time CSCW systems. 210 p. Yhteenveto 2 p. 2003.
30 Markkula, Jouni, Geographic personal data, its privacy protection and prospects in a location-based service environment. 109 p. Yhteenveto 2 p. 2003.
31 Honkaranta, Anne, From genres to content analysis. Experiences from four case organizations. 90 p. (154 p.) Yhteenveto 1 p. 2003.
32 Raitamäki, Jouni, An approach to linguistic pattern recognition using fuzzy systems. 169 p. Yhteenveto 1 p. 2003.
33 Saalasti, Sami, Neural networks for heart rate time series analysis. 192 p. Yhteenveto 5 p. 2003.
34 Niemelä, Marketta, Visual search in graphical interfaces: a user psychological approach. 61 p. (148 p.) Yhteenveto 1 p. 2003.
35 You, Yu, Situation Awareness on the world wide web. 171 p. Yhteenveto 2 p. 2004.
36 Taatila, Vesa, The concept of organizational competence – A foundational analysis. - Perusteanalyysi organisaation kompetenssin käsitteestä. 111 p. Yhteenveto 2 p. 2004.

37 LYYTIKÄINEN, VIRPI, Contextual and structural metadata in enterprise document management. - Konteksti- ja rakennemetatieto organisaation dokumenttien hallinnassa. 73 p. (143 p.) Yhteenveto 1 p. 2004.
38 KAARIO, KIMMO, Resource allocation and load balancing mechanisms for providing quality of service in the Internet. 171 p. Yhteenveto 1 p. 2004.
39 ZHANG, ZHEYING, Model component reuse. Conceptual foundations and application in the metamodeling-based systems analysis and design environment. 76 p. (214 p.) Yhteenveto 1 p. 2004.
40 HAARALA, MARJO, Large-scale nonsmooth optimization variable metric bundle method with limited memory. 107 p. Yhteenveto 1 p. 2004.
41 KALVINE, VIKTOR, Scattering and point spectra for elliptic systems in domains with cylindrical ends. 82 p. 2004.
42 DEMENTIEVA, MARIA, Regularization in multistage cooperative games. 78 p. 2004.
43 MAARANEN, HEIKKI, On heuristic hybrid methods and structured point sets in global continuous optimization. 42 p. (168 p.) Yhteenveto 1 p. 2004.
44 FROLOV, MAXIM, Reliable control over approximation errors by functional type a posteriori estimates. 39 p. (112 p.) 2004.
45 ZHANG, JIAN, Qos- and revenue-aware resource allocation mechanisms in multiclass IP networks. 85 p. (224 p.) 2004.
46 KUJALA, JANNE, On computation in statistical models with a psychophysical application. 40 p. (104 p.) 2004.,
47 SOLBAKOV, VIATCHESLAV, Application of mathematical modeling for water environment problems. 66 p. (118 p.) 2004.
48 HIRVONEN, ARI P., Enterprise architecture planning in practice. The Perspectives of information and communication technology service provider and end-user. 44 p. (135 p.) Yhteenveto 2 p. 2005.
49 VARTIAINEN, TERO, Moral conflicts in a project course in information systems education. 320 p. Yhteenveto 1p. 2005.
50 HUOTARI, JOUNI, Integrating graphical information system models with visualization techniques. - Graafisten tietojärjestelmäkuvausten integrointi visualisointitekniikoilla. 56 p. (157 p.) Yhteenveto 1p. 2005.
51 WALLENIUS, EERO R., Control and management of multi-access wireless networks. 91 p. (192 p.) Yhteenveto 3 p. 2005.
52 LEPPÄNEN, MAURI, An ontological framework and a methodical skeleton for method engineering – A contextual approach. 702 p. Yhteenveto 2 p. 2005.
53 MATYUKEVICH, SERGEY, The nonstationary Maxwell system in domains with edges and conical points. 131 p. Yhteenveto 1 p. 2005.
54 SAYENKO, ALEXANDER, Adaptive scheduling for the QoS supported networks. 120 p. (217 p.) 2005.
55 KURJENNIEMI, JANNE, A study of TD-CDMA and WCDMA radio network enhancements. 144 p. (230 p.) Yhteenveto 1 p. 2005.
56 PECHENIZKIY, MYKOLA, Feature extraction for supervised learning in knowledge discovery systems. 86 p. (174 p.) Yhteenveto 2 p. 2005.
57 IKONEN, SAMULI, Efficient numerical methods for pricing American options. 43 p. (155 p.) Yhteenveto 1 p. 2005.
58 KÄRKKÄINEN, KARI, Shape sensitivity analysis for numerical solution of free boundary problems. 83 p. (119 p.) Yhteenveto 1 p. 2005.
59 HELFENSTEIN, SACHA, Transfer. Review, reconstruction, and resolution. 114 p. (206 p.) Yhteenveto 2 p. 2005.
60 NEVALA, KALEVI, Content-based design engineering thinking. In the search for approach. 64 p. (126 p.) Yhteenveto 1 p. 2005.
61 KATASONOV, ARTEM, Dependability aspects in the development and provision of location-based services. 157 p. Yhteenveto 1 p. 2006.
62 SARKKINEN, JARMO, Design as discourse: Representation, representational practice, and social practice. 86 p. (189 p.) Yhteenveto 1 p. 2006.
63 ÄYRÄMÖ, SAMI, Knowledge mining using robust clustering. 296 p. Yhteenveto 1 p. 2006.
64 IFINEDO, PRINCELY EMILI, Enterprise resource planning systems success assessment: An integrative framework. 133 p. (366 p.) Yhteenveto 3 p. 2006.
65 VIINIKAINEN, ARI, Quality of service and pricingin future multiple service class networks. 61 p. (196 p.) Yhteenveto 1 p. 2006.
66 WU, RUI, Methods for space-time parameter estimation in DS-CDMA arrays. 73 p. (121 p.) 2006.
67 PARKKOLA, HANNA, Designing ICT for mothers. User psychological approach. – Tieto- ja viestintätekniikoiden suunnittelu äideille. Käyttäjäpsykologinen näkökulma. 77 p. (173 p.) Yhteenveto 3 p. 2006.
68 HAKANEN, JUSSI, On potential of interactive multiobjective optimization in chemical process design. 75 p. (160 p.) Yhteenveto 2 p. 2006.
69 PUTTONEN, JANI, Mobility management in wireless networks. 112 p. (215 p.) Yhteenveto 1 p. 2006.
70 LUOSTARINEN, KARI, Resource , management methods for QoS supported networks. 60 p. (131 p.) 2006.
71 TURCHYN, PAVLO, Adaptive meshes in computer graphics and model-based simulation. 27 p. (79 p.) Yhteenveto 1 p.
72 ZHOVTOBRYUKH, DMYTRO, Context-aware web service composition. 290 p. Yhteenveto 2 p. 2006.

73 KOHVAKKO, NATALIYA, Context modeling and utilization in heterogeneous networks. 154 p. Yhteenveto 1 p. 2006.
74 MAZHELIS, OLEKSIY, Masquerader detection in mobile context based on behaviour and environment monitoring. 74 p. (179 p). Yhteenveto 1 p. 2007.
75 SILTANEN, JARMO, Quality of service and dynamic scheduling for traffic engineering in next generation networks. 88 p. (155 p.) 2007.
76 KUUVA, SARI, Content-based approach to experiencing visual art. - Sisältöperustainen lähestymistapa visuaalisen taiteen kokemiseen. 203 p. Yhteenveto 3 p. 2007.
77 RUOHONEN, TONI, Improving the operation of an emergency department by using a simulation model. 164 p. 2007.
78 NAUMENKO, ANTON, Semantics-based access control in business networks. 72 p. (215 p.) Yhteenveto 1 p. 2007.
79 WAHLSTEDT, ARI, Stakeholders' conceptions of learning in learning management systems development. - Osallistujien käsitykset oppimisesta oppimisympäristöjen kehittämisessä. 83 p. (130 p.) Yhteenveto 1 p. 2007.
80 ALANEN, OLLI, Quality of service for triple play services in heterogeneous networks. 88 p. (180 p.) Yhteenveto 1 p. 2007.
81 NERI, FERRANTE, Fitness diversity adaptation in memetic algorithms. 80 p. (185 p.) Yhteenveto 1 p. 2007.
82 KURHINEN, JANI, Information delivery in mobile peer-to-peer networks. 46 p. (106 p.) Yhteenveto 1 p. 2007.
83 KILPELÄINEN, TURO, Genre and ontology based business information architecture framework (GOBIAF). 74 p. (153 p.) Yhteenveto 1 p. 2007.
84 YEVSEYEVA, IRYNA, Solving classification problems with multicriteria decision aiding approaches. 182 p. Yhteenveto 1 p. 2007.
85 KANNISTO, ISTO, Optimized pricing, QoS and segmentation of managed ICT services. 45 p. (111 p.) Yhteenveto 1 p. 2007.
86 GORSHKOVA, ELENA, A posteriori error estimates and adaptive methods for incompressible viscous flow problems. 72 p. (129 p.) Yhteenveto 1 p. 2007.
87 LEGRAND, STEVE, Use of background real-world knowledge in ontologies for word sense disambiguation in the semantic web. 73 p. (144 p.) Yhteenveto 1 p. 2008.
88 HÄMÄLÄINEN, NIINA, Evaluation and measurement in enterprise and software architecture management. - Arviointi ja mittaaminen kokonais- ja ohjelmistoarkkitehtuurien hallinnassa. 91 p. (175 p.) Yhteenveto 1 p. 2008.
89 OJALA, ARTO, Internationalization of software firms: Finnish small and medium-sized software firms in Japan. 57 p. (180 p.) Yhteenveto 2 p. 2008.
90 LAITILA, ERKKI, Symbolic Analysis and Atomistic Model as a Basis for a Program Comprehension Methodology. 321 p. Yhteenveto 3 p. 2008.
91 NIHTILÄ, TIMO, Performance of Advanced Transmission and Reception Algorithms for High Speed Downlink Packet Access. 93 p. (186 p.) Yhteenveto 1 p. 2008.
92 SETÄMAA-KÄRKKÄINEN, ANNE, Network connection selection-solving a new multiobjective optimization problem. 52 p. (111p.) Yhteenveto 1 p. 2008.
93 PULKKINEN, MIRJA, Enterprise architecture as a collaboration tool. Discursive process for enterprise architecture management, planning and development. 130 p. (215 p.) Yhteenveto 2 p. 2008.
94 PAVLOVA, YULIA, Multistage coalition formation game of a self-enforcing international environmental agreement. 127 p. Yhteenveto 1 p. 2008.
95 NOUSIAINEN, TUULA, Children's involvement in the design of game-based learning environments. 297 p. Yhteenveto 2 p. 2008.
96 KUZNETSOV, NIKOLAY V., Stability and oscillations of dynamical systems. Theory and applications. 116 p. Yhteenveto 1 p. 2008.
97 KHRIYENKO, OLEKSIY, Adaptive semantic Web based environment for web resources. 193 p. Yhteenveto 1 p. 2008.
98 TIRRONEN, VILLE, Global optimization using memetic differential evolution with applications to low level machine vision. 98 p. (248 p.) Yhteenveto 1 p. 2008.
99 VALKONEN, TUOMO, Diff-convex combinations of Euclidean distances: A search for optima. 148 p. Yhteenveto 1 p. 2008.
100 SARAFANOV, OLEG, Asymptotic theory of resonant tunneling in quantum waveguides of variable cross-section. 69 p. Yhteenveto 1 p. 2008.
101 POZHARSKIY, ALEXEY, On the electron and phonon transport in locally periodical waveguides. 81 p. Yhteenveto 1 p. 2008.
102 AITTOKOSKI, TIMO, On challenges of simulation-based globaland multiobjective optimization. 80 p. (204 p.) Yhteenveto 1 p. 2009.
103 YALAHO, ANICET, Managing offshore outsourcing of software development using the ICT-supported unified process model: A cross-case analysis. 91 p. (307 p.) Yhteenveto 4 p. 2009.
104 K OLLANUS, SAMI, Tarkastuskäytänteiden kehittäminen ohjelmistoja tuottavissa organisaatioissa. - Improvement of inspection practices in software organizations. 179 p. Summary 4 p. 2009.
105 LEIKAS, JAANA, Life-Based Design. 'Form of life' as a foundation for ICT design for older adults. - Elämälähtöinen suunnittelu. Elämänmuoto ikääntyville tarkoitettujen ICT tuotteiden ja palvelujen suunnittelun lähtökohtana. 218 p. (318 p.) Yhteenveto 4 p. 2009.

106 VASILYEVA, EKATERINA, Tailoring of feedback in web-based learning systems: Certitude-based assessment with online multiple choice questions. 124 p. (184 p.) Yhteenveto 2 p. 2009.
107 KUDRYASHOVA, ELENA V., Cycles in continuous and discrete dynamical systems. Computations, computer assisted proofs, and computer experiments. 79 p. (152 p.) Yhteenveto 1 p. 2009.
108 BLACKLEDGE, JONATHAN, Electromagnetic scattering and inverse scattering solutions for the analysis and processing of digital signals and images. 297 p. Yhteenveto 1 p. 2009.
109 IVANNIKOV, ANDRIY, Extraction of event-related potentials from electroencephalography data. - Herätepotentiaalien laskennallinen eristäminen EEG-havaintoaineistosta. 108 p. (150 p.) Yhteenveto 1 p. 2009.
110 KALYAKIN, IGOR, Extraction of mismatch negativity from electroencephalography data. - Poikkeavuusnegatiivisuuden erottaminen EEG-signaalista. 47 p. (156 p.) Yhteenveto 1 p. 2010.
111 HEIKKILÄ, MARIKKA, Coordination of complex operations over organisational boundaries. 265 p. Yhteenveto 3 p. 2010.
112 FEKETE, GÁBOR, Network interface management in mobile and multihomed nodes. 94 p. (175 p.) Yhteenveto 1 p. 2010.
113 KUJALA, TUOMO, Capacity, workload and mental contents - Exploring the foundations of driver distraction. 146 p. (253 p.) Yhteenveto 2 p. 2010.
114 LUGANO, GIUSEPPE, Digital community design - Exploring the role of mobile social software in the process of digital convergence. 253 p. (316 p.) Yhteenveto 4 p. 2010.
115 KAMPYLIS, PANAGIOTIS, Fostering creative thinking. The role of primary teachers. - Luovaa ajattelua kehittämässä. Alakoulun opettajien rooli. 136 p. (268 p.) Yhteenveto 2 p. 2010.
116 TOIVANEN, JUKKA, Shape optimization utilizing consistent sensitivities. - Muodon optimointi käyttäen konsistentteja herkkyyksiä. 55 p. (130p.) Yhteenveto 1 p. 2010.
117 MATTILA, KEIJO, Implementation techniques for the lattice Boltzmann method. - Virtausdynamiikan tietokonesimulaatioita Hila-Boltzmann -menetelmällä: implementointi ja reunaehdot. 177 p. (233 p.) Yhteenveto 1 p. 2010.
118 CONG, FENGYU, Evaluation and extraction of mismatch negativity through exploiting temporal, spectral, time-frequency, and spatial features. - Poikkeavuusnegatiivisuuden (MMN) erottaminen aivosähkönauhoituksista käyttäen ajallisia, spektraalisia, aikataajuus - ja tilapiirteitä. 57 p. (173 p.) Yhteenveto 1 p. 2010.
119 LIU, SHENGHUA, Interacting with intelligent agents. Key issues in agent-based decision support system design. 90 p. (143 p.) Yhteenveto 2 p. 2010.
120 AIRAKSINEN, TUOMAS, Numerical methods for acoustics and noise control. - Laskennallisia menetelmiä akustisiin ongelmiin ja melunvaimennukseen. 58 p. (133 p.) Yhteenveto 2 p. 2010.
121 WEBER, MATTHIEU, Parallel global optimization Structuring populations in differential evolution. - Rinnakkainen globaalioptimointi. Populaation rakenteen määrittäminen differentiaalievoluutiossa. 70 p. (185 p.) Yhteenveto 2 p. 2010.
122 VÄÄRÄMÄKI, TAPIO, Next generation networks, mobility management and appliances in intelligent transport systems. - Seuraavan sukupolven tietoverkot, liikkuvuuden hallinta ja sovellutukset älykkäässä liikenteessä. 50 p. (111 p.) Yhteenveto 1 p. 2010.
123 VIUKARI, LEENA, Tieto- ja viestintätekniikkavälitteisen palvelun kehittämisen kolme diskurssia. - Three discourses for an ICT-service development . 304 p. Summary 5 p. 2010.
124 PUURTINEN, TUOMAS, Numerical simulation of low temperature thermal conductance of corrugated nanofibers. - Poimutettujen nanokuitujen lämmönjohtavuuden numeerinen simulointi matalissa lämpötiloissa . 114 p. Yhteenveto 1 p. 2010.
125 HILTUNEN, LEENA, Enhancing web course design using action research . - Verkko-opetuksen suunnittelun kehittäminen toimintatutkimuksen keinoin . 192 p. Yhteenveto 2 p. 2010.
126 AHO, KARI, Enhancing system level performance of third generation cellular networks through VoIP and MBMS services. 121 p. (221 p.). Yhteenveto 2 p. 2010.
127 HÄKKINEN, MARKKU, Why alarms fail. A cognitive explanatory model. 102 p. (210 p.). Yhteenveto 1 p. 2010.
128 PENNANEN, ANSSI, A graph-based multigrid with applications. - Graafipohjainen monihilamenetelmä sovelluksineen. 52 p. (128 p.). Yhteenveto 2 p. 2010.
129 AHLGREN, RIIKKA, Software patterns, organizational learning and software process improvement. 70 p. (137 p.). Yhteenveto 1 p. 2011.
130 NIKITIN, SERGIY, Dynamic aspects of industrial middleware architectures 52 p. (114 p.). Yhteenveto 1 p. 2011.
131 SINDHYA, KARTHIK, Hybrid Evolutionary Multi-Objective Optimization with Enhanced Convergence and Diversity. 64 p. (160 p.). Yhteenveto 1 p. 2011.

132 MALI, OLLI, Analysis of errors caused by incomplete knowledge of material data in mathematical models of elastic media. 111 p. Yhteenveto 2 p. 2011.
133 MÖNKÖLÄ, SANNA, Numerical Simulation of Fluid-Structure Interaction Between Acoustic and Elastic Waves. 136 p. Yhteenveto 2 p. 2011.
134 PURANEN, TUUKKA, Metaheuristics Meet Metamodels. A Modeling Language and a Product Line Architecture for Route Optimization Systems. 270 p. Yhteenveto 1 p. 2011.
135 MÄKELÄ, JUKKA, Mobility Management in Heterogeneous IP-networks. 86 p. (145 p.) Yhteenveto 1 p. 2011.
136 SAVOLAINEN, PAULA, Why do software development projects fail? Emphasising the supplier's perspective and the project start-up. 81 p. (167 p.) Yhteenveto 2 p. 2011.
137 KUZNETSOVA, OLGA, Lyapunov quantities and limit cycles in two-dimensional dynamical systems: analytical methods, symbolic computation and visualization. 80 p. (121 p.) Yhteenveto 1 p. 2011.
138 KOZLOV, DENIS, The quality of open source software and its relation to the maintenance process. 125 p. (202 p.) Yhteenveto 1 p. 2011.
139 IACCA, GIOVANNI, Memory-saving optimization algorithms for systems with limited hardware. 100 p. (236 p.) Yhteenveto 1 p. 2011.
140 ISOMÖTTÖNEN, VILLE, Theorizing a one-semester real customer student software project course. 189 p. Yhteenveto 1 p. 2011.
141 HARTIKAINEN, MARKUS, Approximation through interpolation in nonconvex multiobjective optimization. 74 p. (164 p.) Yhteenveto 1 p. 2011.
142 MININNO, ERNESTO, Advanced optimization algorithms for applications in control engineering. 72 p. (149 p.) Yhteenveto 1 p. 2011.
143 TYKHOMYROV, VITALIY, Mitigating the amount of overhead arising from the control signaling of the IEEE 802.16 OFDMa System. 52 p. (138 p.) Yhteenveto 1 p. 2011.
144 MAKSIMAINEN, JOHANNA, Aspects of values in human-technology interaction design—a content-based view to values. - Ihmisen ja teknologian vuorovaikutussuunnittelun arvoulottuvuudet—sisältöperustainen lähestymistapa arvoihin. 111 p. (197 p.) Yhteenveto 2 p. 2011.
145 JUUTINEN, SANNA, Emotional obstacles of e-learning. 97 p. (181 p.) Yhteenveto 3 p. 2011.
146 TUOVINEN, TERO, Analysis of stability of axially moving orthotropic membranes and plates with a linear non-homogeneous tension profile. 104 p. Yhteenveto 1 p. 2011.
147 HILGARTH, BERND, The systemic cognition of e-Learning success in internationally operating organizations. - Kokonaisvaltainen käsitys e-oppimisen menestyksestä kansainvälisissä organisaatioisssa. 100 p. (181 p.) Yhteenveto 1 p. 2011.
148 JERONEN, JUHA, On the mechanical stability and out-of-plane dynamics of a travelling panel sub-merged in axially flowing ideal fluid. A study into paper production in mathematical terms. - Ideaali virtaukseen upotetun, aksiaalisesti liikkuvan paneelin mekaanisesta stabiilisuudesta ja dynamiikasta. Tutkimus paperintuotannosta matemaattisin käsittein. 243 p. Yhteenveto 3 p. 2011.
149 FINNE, AUVO, Tanzanit - Towards a comprehensive quality meta-model for information systems: Case studies of information system quality modelling in East Africa. 209 p. Yhteenveto 2 p. 2011.
150 KANKAANPÄÄ, IRJA, IT Artefact Renewal: Triggers, Timing and Benefits. 79 p. (164 p.) Yhteenveto 1 p. 2011.