

# THE RATIONALE FOR BUILDING A BUSINESS CASE IN SOFTWARE DEVELOPMENT, A CASE STUDY

Jäntti, Marko

University of Kuopio, Department of Computer Science, P.O.B 1627, FIN-70211  
Kuopio, Finland

*mjantti@cs.uku.fi*

**Abstract.** In many software companies, software engineers and business decision makers live in separate worlds, using their own terminology, decision criteria, and working methods. A major challenge is how to integrate business models and software development models together and adjust them to add value for the organization itself and its customers. Building a business case is one possible way to bridge the gap between business and software engineering and to increase the quality and the profitability of software development. Main empirical findings of this study are that case companies used a software business case to allocate resources between concurrent projects, to support sales and pricing activities and to identify the technical platform of their customers' products. A business case was considered as an important part of the software development process. This paper presents the rationale for using a business case for software projects and IT services. We also present that a business case can decrease the number of service barriers, factors that affect negatively a potential IT service end user. Our results are based on both the empirical data from interviews with three case companies within our research project and the literature review.

# Introduction

In many software companies, software engineers and business decision makers live in separate worlds using their own terminology, decision criteria and working methods. In order to achieve a balance between software engineering decisions and business decisions a lot of work must be done: a business case is a good starting point for developing a communication bridge between the business world and the software engineering world. A business case document can include information about the project's resource requirements, investment costs, revenue projections and market acceptance (Jacobson et al., 1999). Robertson (2004) emphasizes the role of a SGS (Scope, Goals, Stakeholders) model as a basis of a business case. Based on these studies we define a business case as follows: It is a document containing business related information such as critical stakeholders and their success criteria, the scope of the software product or a software service, market analysis and competitors, business opportunities, costs, benefits and risks.

For software companies, the goal of building the business case is to increase quality of system development or IT services; they are able to deploy more profitable software products and services than before, within the schedule and the budget. IT customers can use the business case for example in their application integration projects to compare different technological alternatives and costs. A well-defined business case helps IT customers to buy solutions that better meet their requirements. The use of business case in software projects and in IT service development projects is supported by previous studies e.g. (Boehm and Sullivan, 2000; OGC(1), 2002; Reifer, 2002; Robertson, 2004). According to Robertson (2004) a business case provides a mechanism for justifying and guiding the project and helps to answer questions such as "Which requirements are most relevant for the desired benefits?" and "Which design decisions contribute the most value?".

IT Infrastructure Library (ITIL) deals with the best practices for IT service management (OGC(2), 2002). Related to this study, ITIL gives guidance for IT organizations how to perform financial management for IT services and provides a list of issues that a business case for an IT service should include. IT service is a broad concept that needs a more detailed definition. In this study, we use ITIL's definition for a service: "A service is one or more IT systems which enable a business process" (OGC(2), 2002). In practice, IT services can be customer tailored software projects by system integrators, ASP services, consultation, training and hosting services etc. (Rajala et al., 2001).

In their work "Software Economics, a Roadmap " Boehm et al. (2000) argue that design decisions of software engineers are often made in an economics-independent "Flatland" and that there is a "gap" between the value creation criteria and the decision criteria of software engineering. Software engineers do not seem to understand or are not involved in organization's value creation objectives. Many software engineers act like "creative artists" that are only interested in creating

new functions for applications. They do not like business cases and calculations in any form. An example of the "gap" is presented later in this paper. Some studies point out that the gap exists between software developers and customers that use difficult business terms (McDavid, 2000). It is a real challenge to get coders and designers to read, understand and follow business cases. Establishing an efficient business case includes much more than estimating costs for software products. It is a combination of creative thinking, comparing alternatives for technological solutions and continuous updating of stakeholders' needs.

What is the interface between the business and the software development? In previous studies has been argued that the software development process should be integrated with the business process (Bygstad, 2004; Wallin et al., 2002). There have been propositions how to combine business decision models (BDM) and software development lifecycle models (SDLM) for example by mapping Cooper's Stage-Model with Unified Process, Extreme Programming and Microsoft Synchron-and-Stabilize model (Wallin et al., 2002). According to Chillarege (2002) "a good marriage between business dynamics and software engineering process arises from a good match between market values and process attributes (speed to change, scalability, distributed development etc.)". It has also been argued that software project planning and decision-making are affected by marketplace issues such as revenue, costs and risks (Messerschmitt et al., 2004).

In many studies of software business the focus has been on explaining how important the integration between business models and software engineering models is. There is literature showing potential advantages of the software business case (Reifer, 2001) but an interesting point of research is how companies actually use business cases. In some studies a business analysis has been linked to requirements process to identify opportunities for business improvement (Bustard, 1996; Robertson, 2004). There are also case studies where a business case is used to support the software product line approach (Cohen, 2001) and to support using open source software (Kenwood, 2001) and Software Performance Engineering (Williams and Smith, 2003). In this paper, we firstly describe the pattern of a business case as a theoretical rationale why software companies should use the business case in traditional software development. In addition, we present ideas how a business case could be used for building IT services and discuss how a business case is related to Push and Pull factors of services and service barriers. Secondly, we compare the empirical pattern based on the multiple case study with a theory-based pattern.

The rest of the paper is structured as follows: The second section describes the research method used in this study. Third section presents a descriptive theory of the business case in a form of theoretical rationale. In addition it describes how a business case could be used for building IT services. The fourth section presents findings from interviews with case companies and finally in the discussion part our findings are analyzed in more detail and conclusions are presented.

## Methods

This multiple case study is a part of the work of an ongoing research project SOSE (Service Oriented Software Engineering) at the University of Kuopio, Finland. SOSE is funded by the National Technology Agency TEKES. One objective of the SOSE project is to research methods for improving the quality of software development. This paper is a part of above mentioned research and also related to an ongoing doctoral thesis work in the Department of Computer Science. The research consists of four phases:

- How to improve the quality of software development through a business case
- How to improve the quality of software development through service level management
- How to improve the quality of software development through systematic testing (for example by using UML based test model)
- How to improve the quality of software development through a systematic problem management

This paper focuses on the first phase, improving the quality of software development through a business case. The goals of this study included an examination of a) the role of a business case in traditional software engineering and b) the role of a business case for IT services. According to Yin (1989) case studies can be categorized into exploratory, explanatory and descriptive case studies. A descriptive approach was used in this study. The multiple case study method was selected in order to produce theoretical replication.

At first, we present a descriptive theory of business cases and develop a pattern of a business case with a help of the rationale that consists of four questions: what is a business case, why is it needed, when is it established, and who are the potential users of the business case. Our pattern includes following propositions: we propose that 1) organizations do not use a business case frequently in software projects, 2) a business case is rarely a part of the organizations' normal software development lifecycle and 3) in case the business case is used in organization, main reasons to use it are to define critical stakeholders and their success criteria, the scope of the software product or a software service, market analysis and competitors, and to identify business opportunities, costs, benefits and risks.

Secondly, we use a pattern-matching analysis to compare the empirical pattern derived from case studies with a theoretical pattern. The pattern-matching method is considered as a useful technique for linking data to the propositions (Campbell, 1975). Units of analysis were three companies SoftA, SoftB and CustomC. The case study included following questions:

- What kind of software development lifecycle model is used by a company?
- Does a company have a separate product development model in addition to the software development life cycle model?

- How does a company define a Business Model ?
- What is a role of a business case in a company?
- What are the advantages of using business cases in a company?
- What is the content of the business case in a company?

Interviews and group discussions were the main source of evidence (Yin, 1989) in this study. The qualitative data was collected 1) in structured interviews (each 2 hours) that were based on a question form and 2) in discussions of monthly research project meetings (2-3 hours) where all case companies had representatives: a product manager from SoftA, a production manager from SoftB, and an IS manager from CustomC. SoftA is a product-oriented software company. SoftB in turn is focused on tailored software projects and CustomC is a large energy company in Finland. SoftA was selected because it represents large companies with a heavy organizational structure. SoftB is a young and dynamic medium-sized software house with a light organizational structure. CustomC was selected as a representative of IT customers. They do not develop but they buy software products from outside the organization. All three companies are located in Finland and SoftA has several international offices.

The question form that was used as a basis for interviews included questions about companies' general goals for the research project, how and why they use business case analysis and how they use information system and software architecture in their work. Because CustomC is not a software company, we had to modify the first and the second question for them: it is obvious that they do not use software development model. In that case, we asked what are the elements or documents of the software development model that are most useful for them. The second question was whether they use a product development model for example in launching new energy products for customers. Persons who participated in interviews were mainly same people that participated in research project meetings. A researcher's role in project meetings was to participate in discussions and to record the results of discussions.

## Rationale for using a business case

In this section, we present a theoretical rationale for using a business case in software development. We use this rationale to develop a pattern of a business case. There are numerous propositions for the content, the scope and the purpose of the software business case. The studies of Boehm et al. (2000), Cooper (2001) and Robertson (2004) are some good examples of them. Our rationale focuses on four questions: **what** is a business case, **why** is it needed, **when** is it established, and **who** are the potential users of the business case.

**What:** A business case is a collection of business related information that can be used to define critical stakeholders and their success criteria, the scope of the software product, a service or a project, market analysis and competitors, business opportunities, costs, benefits and risks. The business case can be implemented as a software document (for example a business vision document in Unified Process) or as a part of other documents such as in a project management plan.

In order to clarify what is a business case we created a visual model of the role of the business case in software development (see Figure 1). We define that all requirements (functional and quality requirements) from the customer are at first business requirements. A business analyst collects all relevant business requirements, descriptions of customer's business processes into a business case document. A business decision maker, for example a production manager supplements the business case document with e.g. a market analysis, a cost & benefit analysis and information about direct and indirect competitors. A software designer can provide a business case with his/her view on technology choices.

In a next phase a business analyst and a software designer translate business requirements into software requirements using use cases of UML (Jacobson et al., 1999). Later in software projects UML use cases are useful in identifying reusable functional modules for a new system, and deriving and organizing test cases (Jääntti, 2004). The business case should be an open document for all software project members and it should remind workers during the project of the things that create value for a customer. Finally, the software and documents are deployed to the customer including in most cases installation and training services. The customer checks whether the product or a project has met all required business requirements and transfers the payment to the software company's account.

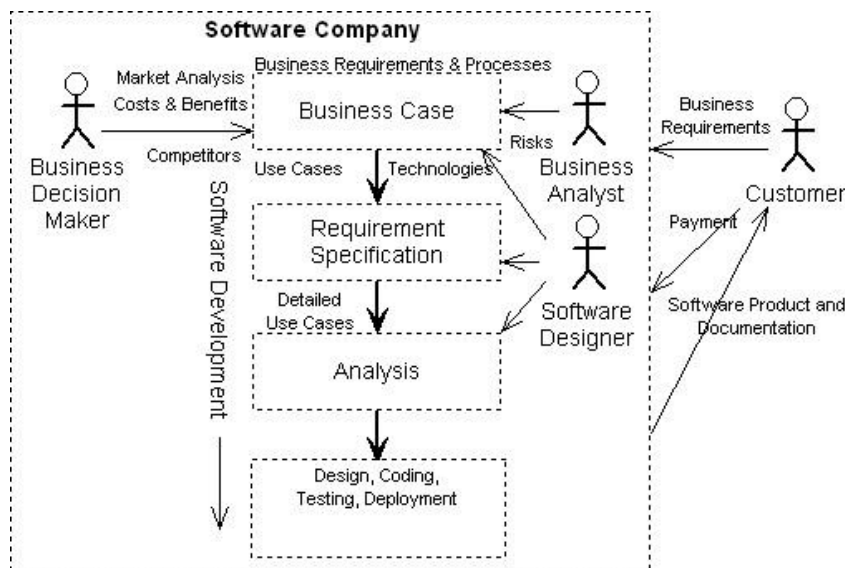


Figure 1. The role of a business case in software development projects.

**Why:** Table I includes describes potential benefits of a business case for a software development project and some contexts where business cases have been used.

<b>Research</b>	<b>Argument</b>
Bustard, 1996	A business analysis is the first step in establishment of requirements, it is used to identify opportunities for business improvement
Jacobson et al., 1999, p. 67	A business case defines the project's resource requirements, investment costs, revenue projections and market acceptance
Boehm and Sullivan, 2000 Boehm, 2003	A business analysis is needed to create a bridge between technology-oriented software engineers and strategy-oriented business decision makers
Cohen, 2001	A business case is used to support the software product line approach
Cooper, 2001	A business case includes market analysis, identifies competitors and users' needs
Kenwood, 2001	A business case is used to evaluate open source software and development methodologies
Wallin et al., 2002	A business case combines business decisions with software development decisions
Reifer, 2001 Reifer, 2002	Important elements of the software business case are return-on-investment, break-even, and cost/benefit calculations A business case can be used to justify the use of agile methods
OGC(1), 2002	A business case includes a financial analysis: budgeting, IT accounting and charging of IT services
OGC(2), 2002	A business case defines what levels of service are expected from the new services, components or technology
Williams and Smith, 2003	A business case is used to justify investing in Software Performance Engineering (reducing costs of performance failures)
Robertson, 2004	A business case helps to find right business requirements and to identify critical stakeholders, it defines the scope of the product or/and the project

Table I: Potential benefits of business cases and usage contexts

A business case document should include business requirements and descriptions of business processes or references to them because they create the basis for further project work: requirement engineering, analysis and design etc. The standard UML notation can be used as a modeling tool; activity diagrams for business processes and use cases for business requirements. According to Demirörs et al. (2003) "business process based approaches are helpful in identifying business requirements as the knowledge is captured at a more abstract, business need level than many other specification approaches". In terms of IT services, a business case can help IT service providers to decrease the number of service barriers and to increase the quality of IT services. A separate business case might not be needed in projects where business requirements and goals are clearly defined, the technology has been selected and a cost-benefit analysis has been performed. However, we recommend that it should be used in all projects. A lack

of the business case might lead to a situation where a system implemented in a project does not support organization's business processes at all or causes massive development costs.

**When:** A business case is usually established before the requirement specification phase in the waterfall model and in the inception phase of the Rational Unified Process. In the Cooper's Stage-Gate process model, building a business case is done before a development stage and is one of the most important stages (Cooper, 2001). Gilpert (2004) has argued that a discovery phase is a key to the success of any software project. In other product development models or in software development lifecycle models the position of the business case could be in stages called exploration, preliminary analysis etc.

**Who:** The business case might be used by business decision makers, business analysts, project managers, IT service managers, software designers, and product managers etc.

## The role of the business case for IT Services

This section examines in more detail the role of a business case in building new IT services. Our objective is to present new viewpoints on IT service management. A business case might be useful in guiding and justifying the development of a new service:

First, the business case for a new IT service helps organizations in budgeting, accounting and charging services (OGC(1), 2002). Budgeting is responsible for issues such as how much money is needed to run the service (for example annual costs of a service), comparing actual costs of the service with the predicted costs and reducing the risk of failing to meet budget. Accounting helps IT organizations in ROI (Return-on-Investment) or cost-benefit analyses and identifying the costs of implementing changes. Charging identifies how service costs can be charged from the customer and gives an opportunity to affect user and customer behavior (reduce the inefficient use of resources by customers). The business case for a new IT service or a software product should be checked several times also during the development phase. If the budget has been exceeded, the project manager should find out the amount of overspending and the reasons for unexpected costs.

Second, the business case guides organizations to decide whether it is more cost-effective to buy software licenses or rent an application from application service providers. More and more traditional software product vendors have started to provide their services via the Internet (Hoch et al., 1999). Advantages of renting services are that there is no need to make any software installations and a service provider is responsible for maintenance costs.

Third, we propose that a business case can be used to define suitable PUSH and PULL factors for a service and to minimize service barriers (see Figure 2). Based on discussions with case organizations in research project meetings and the

ITIL framework (OGC(1), 2002), we developed a model where a potential service end user is affected by PUSH and PULL factors and service barriers. In further research, there is a need to create a between our model and commonly used theories of predicting social behavior (Ajzen et al. 1980). *PUSH factors* are factors that push potential end users to use services. *PULL factors* are factors that attract potential end users to use services and *service barriers* are factors that prevent an end user from using a service and achieving the desired goal or that slow down the process of using a service.

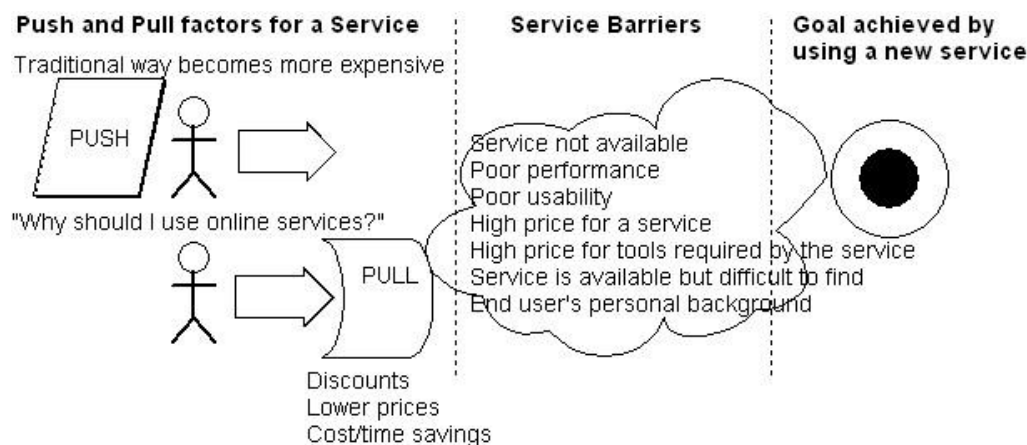


Figure 2. Service barriers and Push and Pull factors.

One example of Push factors is that an end user must pay a higher price for doing things in a traditional way. Most banks in Finland have increased payments for customers that walk in to the bank and deal with their issues with a bank officer. Online banking services in turn are provided without a charge or with a minimum charge. This is an efficient way to push end users to use new services. Pull factors are typically discounts and lower prices for products and services if an end user chooses to use online services. Next, we discuss in more detail how a business case helps to avoid or minimize following service barriers:

a) Unavailability of services and poor performance: a business case should define what levels of service are expected from the new services, components or technology. A service provider's solution for this problem is to establish and monitor *service level agreements*, contractual specifications of service standards (OGC(1), 2002).

b) Poor usability: A business case should record most important requirements of end users such as usability requirements. Some online services use the terminology that is difficult to understand by normal end users. Some services require end users to enter information that they cannot remember such as online electricity agreement services that need information about the size of the fuse

(safety device used to protect an electric circuit against an excessive current) used in apartment. A typical user does not remember the size of the fuse and the process of making an online electricity agreement will slow down or discontinue. As a conclusion, usability testing plays an important role in designing new online services.

c) Too high price for services or tools required by a service: Customers may think that a service or new equipment needed in a service is too expensive for them. Mobile services are good examples here. In order to use new mobile services provided by operators, an end user is often forced to buy a new and an expensive mobile phone that supports a service. A business case should identify the means (pricing strategies and charging options, for instance) that attract customers try the service or the product at least once. IT service providers often provide for example free service trials. A business case should contain a list of benefits of the service for customers that compensate the high price.

d) A service is difficult to find: A service provider has implemented a high quality service with expensive usability tests, the service is available 24 hours per day and 7 days in a week and is provided free but nobody uses it, why? The reason could be in marketing of service. A new service needs marketing as all products do. Both customers and company workers need to be informed of new services. IT organizations need a service catalogue (OGC(1), 2002) that includes both the services visible to customers and company's internal services that are not visible to customers.

e) End user's personal background: Personal background of an end user (lack of confidence in online services, age, computer skills, disabilities) is probably one of the most common service barriers. Many people are still scared of security threats of online banks or avoid Internet payments with a credit card. Old persons who do not have any computer skills will continue using the service desk of the bank although their service charges were much higher than those of online service users.

## Findings from a Case Study

### The role of the business case in traditional software engineering

Three case companies in research project SOSE (Service Oriented Software Engineering) answered the questions that were described in the Method section. Some information related to the business case was also collected in SOSE project meetings. The purpose of three first questions was to identify the interaction between business (product development) models and software development

models. The purpose of the fourth question was to find out how often the business case is used in projects and the last question identified how companies see the advantages of building a business case.

SoftA is a product-oriented software company that uses a combination of the traditional waterfall model and the Microsoft Synchronise-and-Stabilize model as a software development lifecycle model. They did not use any specific product development model in addition to the software development life cycle model but they had their own product development method covering business decisions. The business case helps SoftA in pricing decisions and allocating resources for concurrent projects. The business case is a fixed part of their software development process.

SoftA's business case consisted of a description of microenvironment, a market analysis, a product strategy, a marketing strategy, a risk analysis and some financial issues (e.g. cost analysis and estimated turnover of the product). They defined the business model as follows: "The core of the business model is our product that generates the business. In the future, the focus of our business model will be customers and their needs."

SoftB's core business is focused on implementing software projects. They had a combination of two software development life cycle models: Unified Process and Extreme Programming. They had developed a lightened version of Unified Process for their own needs. There was no need for a separate product development model in the SoftB. As advantages of the business case they mentioned that it identifies the technical platform of customers' products. A software company is able to take account in interoperability issues between customers' legacy systems and a new software module that they want to sell to the customer. The business case of SoftB includes benefits why customers should buy the product/project. The business case is also a fixed part of SoftB's software development process. They defined the business model as follows: "Business model is a model that describes what we produce and how we do it. Probably, our software developers would understand it as business logic (a collection of functional parts in system architecture)."

The SoftB's business case was divided into two documents but included partly the same elements than SoftA's business case: a description of microenvironment, a market analysis, a product strategy, a marketing strategy, and a risk analysis. Development issues are usually described in a project management plan and a financial plan is part of the general business plan.

CustomC is an energy company (specialised in electricity sales and district heating) that buys software products. They do not produce software and do not have a software development model. Architecture descriptions are the most interesting software documents for CustomC at this moment. CustomC has used a software business case in some application integration projects. They considered the risk management part of the business case an interesting area. CustomC stated in a project meeting that software companies should give customers the estimation

of the software product lifecycle because it is an important factor that affects the buying decision. The lifecycle of a cheaper software product is probably shorter than the lifecycle of a more expensive product (implemented with a richer set of interfaces etc.). If the lifecycle of the software product was defined, an IT customer would be better able to estimate the costs of redesign/reengineering of their software modules (for example, in 3 years we have to implement big changes to applications X and Z, in 5 years we may have to abandon applications D, E and F).

A summary of the findings from interviews with SoftA, SoftB and CustomC is presented in Table II.

Question	SoftA	SoftB	CustomC (Customer)
1. Software development lifecycle model	Waterfall model with some features of MS Synch & Stabilize model	Unified Process with some features of Extreme Programming	No software development model is needed. Architecture descriptions are especially important.
2. Need for a separate product development model	Separate product development model exists	No separate product development model is needed	No specific product development model
3. Definition of the business model	The core of the business model is our product that generates business.	The business model is what we do and how we do it.	Our business is to produce and sell energy for our customers
4. A role of the business case	A part of the software development lifecycle model	A part of the software development lifecycle model	A business case is used in some application integration projects
5. Advantages of the business case	Supports pricing and resource allocation, justifies the project	Identifies the customer's technical platform, includes a list of system benefits for the customer	A business case should include estimates of the software lifecycle

Table II: The findings of the interviews with three case companies

An example of the typical gap between software engineers and business decision makers came up in discussions of project meetings. CustomC stated that it might take days or weeks before the software supplier gives the final price for the solution that they would like buy. Software engineers tend to calculate costs of software production very detailed like "an accounting department". They want to calculate the cost of every single piece of material exactly before they can give a total price for a product or a project. Business decision makers (or customers such as company C) appreciate to have rough price estimates as soon as possible to be utilized in negotiations. market. Estimation of work efforts is complicated. That topic is discussed in more detail by Haapio (2004).

## Discussion

This paper presented a theoretical rationale for the use of the business case by answering the following questions: what is a business case, why is it needed, when is it established and who are the potential users of the business case. This rationale was used to develop a pattern of a business case. Because previous studies give support for using a business case in software development, we conducted a multiple case study to explore why and how our case companies use a business case in traditional software engineering. Additionally, we discussed how a business case could be used for building IT services and presented effects of push factors, pull factors and service barriers on a service end user.

Main empirical findings of the pattern-matching analysis were that case companies used the business case mainly for the same purposes than in our theory-based pattern. Additionally, the business case was used to support sales activities and pricing decisions, to allocate resources between concurrent projects within the organization and to identify interoperability issues between customer's product platform and products offered by the software company. The IT customer had used the business case in some application integration projects. The IT customer's viewpoint was that a business case should include some information about the lifecycle of the product. In contrast to our theory-based pattern, we noticed that both software companies considered a business case as a fixed part of the software development model.

We selected three different case companies for this study: a software product company, a tailored software project company and an IT customer. Initially we had five companies as case units but two of them did not participate in interviews because the topic was not current for them. In the future, it would be interesting to conduct a survey about how frequently organizations use business cases in their software projects and how important role a business case plays in improving the quality of software project compared with other project documents. Questions such as who are the persons who write, read or have an access to the business case documents in software companies were omitted in this study but are valuable for further research. Another possibility to extend our research could be to interview the lower organization level of companies (for example system designers or coders) and we could compare those results with our current results where the target group consisted of production managers and R&D managers. We are also going to extend the literature review during the research project.

We have to admit that we were quite disappointed with empirical findings from interviews and project discussions in terms of how companies use business cases. We expected that companies would give us detailed information about their business model in terms of four elements: product development, services, revenue logic, marketing & sales. However, all definitions of the business model were in a very abstract level. Real business case documents would have been useful sources

of evidence but they were not available. One reason for this might be that a business case includes too much confidential information. However, interviews were useful finding the reasons why companies use business cases and motivated us to study the role of business case in building IT services.

The second goal of this study was to examine the role of a business case in building IT services and proposed that a potential service end user is affected by PUSH and PULL factors and service barriers. We proposed that a business case might be used to decrease the number of service barriers such as unavailability of services and poor performance. The goal of the second part was to present new viewpoints on IT service management. Further work related to this topic is to research the effect of service barriers on different online services more thoroughly and link our model to theories that deal with e.g. predicting social behavior.

The main contribution of this study is that it increases understanding of importance of building a business case and gives a general overview about the current methods of software development and software business in software companies that belong to our research project. The rationale we presented in this paper will definitely evolve in the course of our research project. Empirical results of this study and theory results supplement each other.

The business case adds value for both software companies and their customers. As a well-designed document it will probably increase the quality of system development but further research is needed to prove this. Additionally, this study clarified our research goals and opened the communication link between the research group and case companies in the research project. As a conclusion we state that the business case in software development is a very broad and an evolving concept. There are numerous propositions for the content, the scope and the purpose of the software business case. We consider building a business case as an attractive way to increase the quality of system development.

## Acknowledgments

This paper is based on research in the SOSE project (2004-2006), funded by TEKES (the National Technology Agency), European Regional Development Fund (ERDF), ICT and customer companies in electricity domain. I wish to thank professor Anne Eerola for her comments, researcher Harri Karhunen for his help in data collection and companies for participating in interviews.

## References

- Ajzen, I. and Fishbein, M. (1980). *Understanding Attitudes and Predicting Social Behavior*, Englewood Cliffs, NJ: Prentice-Hall, Inc.
- Boehm, B. and Sullivan, K. (2000). Software economics: a roadmap. *Proceedings of the conference on The future of Software engineering*.
- Boehm, B. (2003). Value-based software engineering. *ACM SIGSOFT Software Engineering Notes*, Volume 28, Issue 2.
- Bustard, D.W., Dobbin, T.J. and Carey, B.N. (1996). Integrating Soft Systems and Object-Oriented Analysis. *Proceedings of the 2nd International Conference on Requirements Engineering (ICRE '96)*, pp. 52.
- Campbell, D. (1975). Degrees of freedom and the case study. *Comparative Political Studies*, 8, 178-185.
- Chillarege, R. (2002). The Marriage of Business Dynamics and Software Engineering. *IEEE Software*, pp.43-49.
- Cohen, S. (2001). Case Study: Building and Communicating a Business Case for a DoD Product Line. *Technical Note, CMU/SEI-2001-TN-020*, Software Engineering Institute.
- Cooper, R.G. (2001). *Winning at New Products*. 3rd ed., Perseus Publishing, Cambridge, Mass.
- Demirörs, O. Gencil, Ç. and Tarhan, A. (2003). Utilizing Business Process Models for Requirements Elicitation. *Proceedings of the 29th Euromicro Conference (EUROMICRO'03)*, pp. 409.
- Gilpert, S. (2004). Wearing Two Hats: Analyst-Managers for Small Software Projects. *IT Professional*. July/August 04, pp. 34-39.
- Haapio, T. (2004). The Effects of Non-Construction Activities on Effort Estimation. In *Electronic Proceedings of the 27th Information Systems Research in Scandinavia (Proceedings of the IRIS'27)*, Falkenberg, Sweden, August 14-17, 2004, p.[13].
- Hoch, D.J., Roeding, C.R, Purkert, G. and Lindner, S.K. (1999). *Secrets of Software Success - Management Insights from 100 Software Firms around the World*. Harvard Business School Press, Boston (MA).
- Jacobson, I., Booch, G. and J. Rumbaugh (1999). *The Unified Software Development Process*. Addison-Wesley.
- Jääntti, M. and Toroi, T. (2004). UML-based Testing: A Case Study. *Proceedings of 2nd Nordic Workshop on the Unified Modeling Language* (Turku, Finland, August 19-20, 2004).
- Kenwood, C.A. (2001). A Business Case Study of Open Source Software. *Technical Paper*, July 2001, MITRE, Available online at:  
[http://www.mitre.org/work/tech\\_papers/tech\\_papers\\_01/kenwood\\_software/](http://www.mitre.org/work/tech_papers/tech_papers_01/kenwood_software/)
- McDavid, D. (2000). A Framework for Discovering, Organizing, and Using Business Semantics. *Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 34'00)*, pp. 549.
- Messerschmitt, D.G. and Szyperski, C. (2004). Marketplace Issues in Software Planning and Design. *IEEE Software*, Vol. 21, No. 3, May/June 04, pp. 62-70.
- Office of Government Commerce (1) (2002). *ITIL Service Delivery*. The Stationary Office, UK, Ref. use in text, OGC(1).
- Office of Government Commerce (2) (2002). *ITIL Infrastructure Management*. The Stationary Office, UK, Ref. use in text, OGC(2).

- Rajala, R., Rossi, M., Tuunainen, V. and Korri, S. (2001). Software Business Models: A Framework for Analysing Software Industry. *Tekes, Technology Review* 108/2001, pp. 37-38.
- Reifer, D. J. (2001). *Making the Software Business Case: Improvement by the Numbers*. Addison Wesley.
- Reifer, D. J. (2002). How good are agile methods. *IEEE Software*, July/August 2002.
- Robertson, S. (2004). Requirements and the Business Case. *IEEE Software*, September/October 04, pp. 93-95.
- Wallin, C., Ekdahl, F. and Larsson, S. (2002). Integrating Business and Software Development Models, *IEEE Software*, November 02, pp. 28-33.
- Williams, L.G. and Smith, C.U. (2003). Making the Business Case for Software Performance Engineering. Software Engineering Research and Performance Engineering Services, Available online: <http://www.perfeng.com/papers/buscase.pdf>.
- Yin, R. K. (1989). *Case study research: Design and methods* (Rev. ed.). Beverly Hills, CA: Sage Publishing..