



UNIVERSITY OF GOTHENBURG

Situational Factors Affecting Software Development Process Selection

**BURÇIN DEDE
IRINA LIOUFKO**

Thesis work for Master of Science in Software Engineering and Management

**Report No. 2010:105
ISSN: 1651-4769**

Table of Contents

ABSTRACT	2
1. INTRODUCTION	2
1.1 Background	2
1.2 Purpose	2
1.3 Research Question	3
1.4 Disposition	3
2. RESEARCH APPROACH	3
2.1 Research Strategy	3
2.2 Data Collections	3
2.3 Data Analysis	4
3. THEORETICAL BACKGROUND	4
3.1 Process Assessment	4
3.1.1 What to gauge?	4
3.1.2 How to gauge?	5
3.2 Agility	6
3.3 AgileForUs	6
3.3.1 Success Factors	6
4. RESULTS	7
4.1 Process Assessment Survey	7
4.2 Situational Factors Categories	7
4.3 Answer to Research Question	9
5. ANALYSIS AND DISCUSSION	9
5.1 Related Work	9
5.2 Interviews Results	9
5.3 Limitations and Future Work	10
6. CONCLUSIONS	10
7. ACKNOWLEDGEMENTS	10
8. REFERENCES	10
Appendix 1	13
Appendix 2	14
Appendix 3	15
Appendix 4	16

Situational Factors Affecting Software Development Process Selection

Burçin Dede
University of Gothenburg
Gothenburg Sweden
0737695790
burcin_dede@yahoo.com

Irina Lioufko
University of Gothenburg
Gothenburg Sweden
+46 739 943 776
irina.lioufko@gmail.com

ABSTRACT

Context: Within organizations which provide IT, R&D and consulting services to diverse markets, it is common to use different software development processes. It helps organizations to meet the requirements of projects, which focus on different business domains. Herein the challenge is to determine which factors should be considered by project managers, when it comes to choosing appropriate process for each particular project.

Objectives: This study aims to evaluate the gap between acquired and deployed technology for selective agile process implementations and identify situational factors (SFs), which influence process selection within organizations.

Method: The qualitative semi-structured and spontaneous, informal interviews, literature review, quantitative web-based questionnaire and validation meetings with project managers were used to define and collect necessary data and validate thesis results

Results: The result of this thesis work is presented as a table, containing 17 situational factors, which have been identified and grouped under 6 main categories. Moreover, survey results provide adoption rate per every investigated process, serving also as “evidence of match”, proving that right process was chosen for each particular project within companies.

Conclusions: It is concluded that it is important to be aware and take into account SFs to get the optimal match between each project-process pair. It can help to increase adoption rate of the process, which was deployed, and use maximum of the benefits, tailored to the specific process.

Keywords

Situational factors, Assimilation gap, Software Process Improvements, Software Development Process, Process Adoption, Process Assessment.

1. INTRODUCTION

1.1 Background

During the last decades the speed of change in the software industry has increased (Borjesson, 2006a). To be successful, software organizations must increasingly be organized, managed, and executed in ways that allow them to effectively sense and respond to unpredictable events in their environment (Rogers, 2003). Thus, software process improvement becomes a central and critical success factor for the development and consolidation in competitive software industry. Furthermore, agile software development practices, being both effective and flexible, are believed to help organizations to be nimble and responsive enough to meet different needs of market.

Software development scenarios are varying from project to project; and accordingly, project managers need different software development processes to deliver various services within a company.

The usage of optimal process in each particular project can minimize the failure risk, reduce waste and make more profit. Therefore, methods for selection are critical factors for a project to be successful. Moreover, as no software project can be handled exactly same way, the success of each project depends on different combination of environmental factors. For instance, while some projects can be accomplished within ten years, others can be done just in ten days. These factors make the task of managing software projects extremely complex within a company (Rose, Pedersen, Hosbond and Kraemmergaard, 2007).

1.2 Purpose

This thesis is performed at the company ConsultIt, IT services provider in Western Europe, with subsequent additional involvement of East European development company ItSystems. ItSystems is a small company and use agile process called “Common Sense” for almost all projects. ConsultIt serves large and medium-sized organizations of diverse markets with customers in variety of business domains from telecom to healthcare. ConsultIt project managers are aware that various projects require

different development process. Therefore, there are different types of software development process at the company.

In this paper, we aim to investigate how organization’s current processes could be measured to identify how flexible and agile they are, and also to analyze situational factors that influence process selection within organization as possible means of achieving increased organizational success.

“A situational factor (SF) is any factor relevant for product development and product services. Examples are company size, branch and the number of submitted requirements per month, whether or not currently a waterfall-based method is used for product software development, etc.” (Weerd, Brinkkemper, Nieuwenhuis, Versendaal, & Bijlsma, 2006).

The purpose of this study is to find out situational factors which affect the selection of development process within organizations. Key interests are to determine the assimilation gap (Fichman & Kemerer, 1999) for processes, and analyze selected software process implementations to identify situational factors.

1.3 Research Question

In this paper, three agile process implementations – “AgileForUs”, SCRUM and “Common Sense” are analyzed. While SCRUM is a well-known agile process, AgileForUs and Common Sense are only used by specific companies, where they have been developed.

The *Research Question (RQ)* addressed in this thesis is: “What situational factors should an organization look for when choosing among AgileForUs, SCRUM and Common Sense?”

1.4 Disposition

The rest of this paper is organized as follows. Section 2 describes the research approach and data collection methods used. Section 3 presents theoretical background of the paper, introducing process assessment and new agile process - AgileForUs, which has been recently developed at ConsultIt. Section 4 presents the results of the study by providing the analysis of the employee process assessment survey and interviews. Findings are then discussed and related to theoretical background in Section 5. Related work, limitations and future work is also outlined here. The last section (section 6) summarizes the conclusion of the study.

2. RESEARCH APPROACH

The research strategy and data collection design is presented in this section.

2.1 Research Strategy

In this paper, multiple case study (Yin, 1994) was used as a primary research method to evaluate the value gap for the projects using different software development processes and to analyze situational factors as influential parameters

for choosing a particular software development process (SDP). Stake (2000) argues that collective case study, a third type of case study within qualitative context, is useful to investigate a particular phenomenon or general condition by collecting data from multiple cases. Quantitative data is collected as a supplement to qualitative data, which provides better understanding of rationale or theory underlying relationships (Soy, 1997).

The required data has been collected using both qualitative and quantitative methods to achieve stronger research design, and more valid and reliable findings. The benefit of using multiple methods also provides us validation of data through cross verification from more than two sources (Bogdan, R. C. & Biklen, 2006).

2.2 Data Collections

Table 1 below summarizes the data sources we use throughout this study. Interviews are used as qualitative data sources, whereas surveys are used as a quantitative source. Web-based survey and interview transcripts analysis have been done for first two cases. In the third case, notes from the interview have been analyzed together with the data from conducted survey. The number of survey respondents is written in parenthesis along with each survey.

	Process	Company	Data Source
Case 1	AgileForUs	ConsultIt	Interview Transcript
Case 1	AgileForUs	ConsultIt	Survey (4)
Case 2	SCRUM	ConsultIt	Interview Transcript
Case 2	SCRUM	ConsultIt	Survey (6)
Case 3	Common Sense	ItSystems	Notes
Case 3	Common Sense	ItSystems	Survey (8)

Table 1. Data sources

Table 2 below provides an overview of the data collection techniques used in this paper.

Data Collection Method	Explanation
Informal interviews, Spontaneous interviews	One of the authors has conducted informal interviews and discussions with company ItSystems, participating in the study. The authors have had spontaneous interviews and discussions with Håkan Neeman and ConsultIt managers participating in the study.
Semi-structured, Open-ended, Interviews	A set of semi-structured interviews has been performed. These interviews focus on

	positive and negative aspects of process implementation.
Questionnaire, Employee Process Assessment Survey	Web-based surveys have been sent out containing structured questions about appreciation and understanding of the development process followed in the project, its effectiveness and use: 2 open ended questions and 12 close ended questions (six-grade Likert scales).
Literature studies	The appropriate literature was studied discussed and reflected upon, providing knowledge on SPI in general, activities and initiatives in particular, Change Management Models
Internal Consult/It Process Documentation Review	Review of Process Descriptions and Presentations, Project Decisions, Final Reports

Table 2. Data collections

2.3 Data Analysis

As Data Analysis method we use Noticing, Collecting and Thinking Process (NCT), which was developed Seidel in 1998 (Seidel, J. V., 1998). This qualitative data analysis (QDA) method was used to analyze interviews. At the same time, surveys were conducted and evaluated according to the measurement mechanism, which was designed and implemented by Anna Borjesson at Ericsson (Borjesson, 2006a).

Qualitative data analysis process consists of three parts: Noticing, Collecting, and Thinking about the things. The process and the relationships among its parts are represented in Figure 1 below.

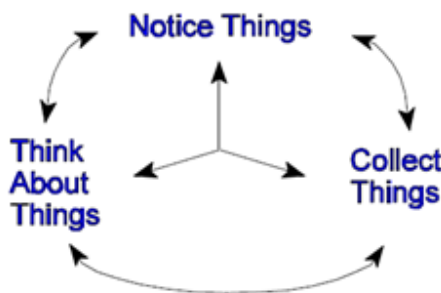


Figure 1. Qualitative Data Analysis (Seidel, 1998)

As it is seen in Figure 1, this QDA process is not linear and has three important characteristics, which should be considered during analysis.

“Iterative and Progressive: When we are thinking about things we also start noticing new things in the data. We then collect and think about these new things. In principle the process is an infinite spiral.

Recursive: While collecting the things, we might simultaneously start noticing new things to collect.

Holographic: Each step in the process contains the entire process. For example, when we first notice things, we are already mentally collecting and thinking about those things.”(Seidel, J. V., 1998)

Initial goals were determined and research question was formulated during “noticing” part. During “collecting” phase interview questions were prepared and semi-structured interviews were conducted. During “thinking” part initial goals were reexamined based on analysis of data gathered in “collecting” phase.

In this study, all three phases (“noticing”, “collecting”, and “thinking”) were done iteratively while conducting interviews with project managers. After the interviews were hold, the interview transcripts and additional questions were sent to the interviewees to get their feedback and remarks. NCT processes were followed until the final validation of the thesis results.

3. THEORETICAL BACKGROUND

3.1 Process Assessment

Knowledge about SPI is one key factor to successful SPI initiatives and there exists a vast body of knowledge about Change Management and SPI in the field today.

(A simple search on <http://scholar.google.com> returns around 76 000 articles on “Change Management” and more than 10 000 articles on “Software Process Improvement”)

Many of the known researchers in SPI literature agree on the necessity of measuring to understand and improve practice (Humphrey, 1989; McFeeley, 1996; Grady, 1992; 1997; Weinberg, 1993; Zahran, 1997). It has been shown that measuring SPI progress is an important key factor to SPI success.

Moreover as Fayad and Laitinen (1997) state, the first step in any software process improvement program must be assessment, because assessment is supposed to give an organization a sense of where it stands.

3.1.1 What to gauge?

Fichman & Kemerer (1999) have identified a gap which they call the assimilation gap (Figure 2); this gap illustrates the difference between acquired and deployed technology, e.g. software process, in acquiring organizations. They argue that it is possible for an organization to err by adopting the right process but failing to implement it in a way that generates benefits for the organization.

This gap expresses the organizations’ success/failure rate as the difference between the intended practices with the acquired process or system and the actual practices that emerge as a result from the implementation. (Håkan Neeman, 2007)

According to Fichman & Kemerer (1999), it is deployment that makes organizational impact and thus it is more

important to understand the deployment curve than the acquisition curve.

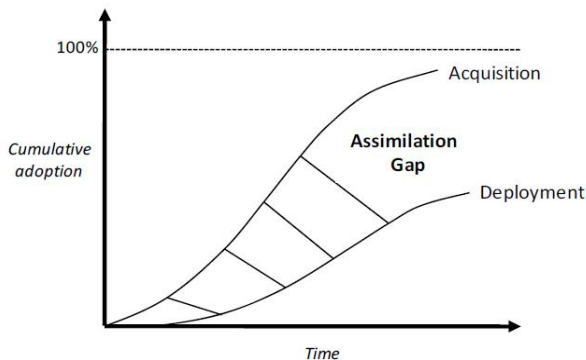


Figure 2. Assimilation Gap (Fichman & Kemerer, 1999)

There is also another research acknowledging assimilation gap. Already in 1978 Argyris and Schön argued there are *espoused theories*, a conception of what one wants to do, and *theories-in-use*, action as actually performed. According to this argument there can be gap between espoused processes and processes-in-use which causes project failures, waste of resources and money within organizations. Therefore, making process assessment and improvement is crucial to be successful in software industry.

3.1.2 How to gauge?

3.1.2.1 SPI Measurements

Measuring SPI is expensive and requires a great deal from people that usually already has a lot to do, argues both Humphrey (1989) and Weinberg (1993). The complex task of measuring is widely known in the software engineering field, and in the literature many warning flags are raised when it comes to measuring SPI.

In most assessment models, the organization evaluates its development capability against a set of “best practices” that are supposed to be found in effective organizations (Fayad and Laitinen, 1997). There are a large number of process improvement initiatives, of which, the best known are the Software Engineering Institute’s Capability Maturity Model (SEI CMM), SPICE, the U.S. Department of Defense SDCE, ISO 9000, and ISO/IEC 12207. Some programs allow self assessment while others require outside certification.

The SEI CMM is one of the best-known and most widely discussed software process improvement models. In the capability maturity model (Paulk *et al.* 1995) we see that even though measurement is an integral part of the model, it is not until level four that the key practices area, measurement, is introduced. To reach this level it is required for an organization to have measurements for productivity and quality for the most important software project activities across all projects as a part of an organizational measurement program (Paulk *et al.* 1995).

However, not many organizations manage to actually reach CMM level four (SEMA, 2002). One explanation for the low success rate in the CMM model might be that many measurement activities in SPI focus on the end results, like increase or decrease in productivity, whereas SPI is a continuous iterative process. This process cannot exist without continuous and systematic monitoring and measurement of an organizations own processes (Zahran, 1997).

As reported by El Emam and Briand, many organizations find the early adoption of certain processes, such as change control and code reviews, more effective than adopting them in the recommended sequence (El Emam and Briand, 1997).

Despite many of the known difficulties when it comes to measuring SPI, we cannot find any literature that argues for a halt in measurements. On the contrary, we must not stop measuring states Humphrey (1989). However, DeVellis (2003) states that if a poor measure is the only one available, the costs of using that measure may turn out to be greater than any benefits attained. This shows the importance of not only measuring SPI progress, but also performing appropriate analysis of the measurements. Without proper measurements and analysis, suggested actions based on SPI outcome are at the risk of being perceived as yet another opinion (Borjesson, 2006b). Yet there exists no comprehensive measurement framework in the SPI literature (Iversen and Ngwenyama, 2006), which complicates matters when it comes to actually measuring SPI progress.

3.1.2.2 ERICSSON Process Model

New knowledge in the SPI measurement field is constantly needed in order for SPI activities carried out within organizations to be successful. As part of contributing to this body of knowledge Borjesson *et al.* (2007) performed research in the area of process innovations and improvement measurement mechanisms (Rocha, 2008).

During a longitudinal action research project between 2001 to 2006 Borjesson *et al.* (2007) designed and implemented a measurement mechanism at a department at Ericsson. In the development of the process model Basili’s (1994) Goal-Question-Metric approach was used and as a result four metrics were developed: Processment, Process Commitment, Process Improvement and Process Learning. Each of these four metrics represents target areas within Ericsson’s SPI work. The survey used to measure the process knowledge and use within Ericsson. At that time it consisted of ten questions. Each of these questions was in turn connected to one of four metrics as can be seen in Figure 3 below. (Rocha, 2008)

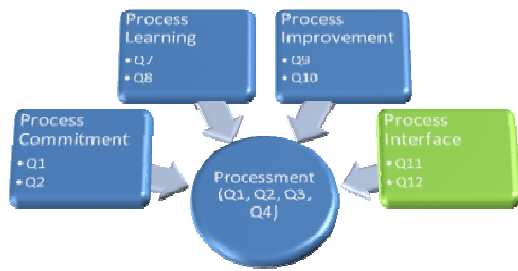


Figure 3. Ericsson Process Model

Continuing the work with the process model, Enskog (2006) conducted a research project at Ericsson where she analyzed the results of a process assessment survey sent out in May 2006 and validated the process model with the data from this survey. (Rocha, 2008)

In the employee process survey sent out in November 2007 two new questions were added, making a total of twelve questions. The two new questions formed a new metric, Process Interface. This key ratio is added to the process model, making a total of five key ratios (see Figure 3).

In the previous research, Ericsson process survey has been mostly conducted within dedicated departments for assessing new technologies and methods, e.g. R&D department.

However, no research has previously been concerned with applying this mechanism for an IT services provider companies implementing IT solutions to customers. Therefore, this study focuses on this perspective. The motivation is to verify if this mechanism has a similar successful impact and results; and discover potential problems and issues it may have.

3.2 Agility

Effective organizational change can be achieved by being agile. Agility is a concept which makes organizations capable of changing quickly and gracefully to survive and succeed in today's business environment (Baskerville, Mathiassen and Pries-Heje 2005). Organizations need to be flexible and adaptable in developing IT solutions to survive during times of uncertainty and turbulence in the business environment. In the case of uncertainty and turbulence, being agile helps organizations to respond quickly and effectively to anticipated and unanticipated changes (Highsmith 2002).

Due to flexibility and effectiveness of agile software development practices, usage of more agile processes helps organizations to be nimble, responsive, and able to hit a moving target-in short. Although, there are different types of software development processes at ConsultIt, recently new agile development process, AgileForUs, has been developed to facilitate the power of agility.

ConsultIt's current processes are assessed to provide list of SFs, which has an effect on the software development process, as guidance for company. Before implementing process assessment, AgileForUs process was examined carefully. Further, its practices and principles were understood quite well.

3.3 AgileForUs

AgileForUs is positioned as a process model, based on set of best practices, which is providing agile way of working for multi-site environment. It is an attempt to maintain quality in the globally outsourced projects with low cost. Despite the fact that the main concept is global working, AgileForUs also focuses on the quality. This is what differs it from ROPES, SCRUM, RUP and XP.

AgileForUs also represents an alternative or improvement to how more agile approach can be deployed to many existing and new software projects. Therefore, the power of AgileForUs comes from customer satisfaction. It is flexible to make changes coming from customers in all process phases. Moreover, additional value is created by customer involvement throughout the process. AgileForUs involves stakeholders in the entire process to make global outsourcing more valuable for all participants.

AgileForUs is suitable for large-scale projects as well as smaller ones. Since AgileForUs is designed for working with global teams, it proclaims that software should be decomposed into sub components or parts. In case of very big project, the responsibility for each component can be assigned to different party, onsite or offshore.

AgileForUs has taken many of its elements from other process models, such as ROPES, SCRUM, RUP and XP. It also combines all those process models' strength within one process model to deliver quick prototype and offer an effective alternative way of monitoring progress.

3.3.1 Success Factors

Education and *awareness* are the keys to success in development of the offshore projects and process productivity. AgileForUs spreads domain knowledge among team members and also makes all information available to all team members. Accordingly, whole team progress awareness is reached. The process model makes stress on both technical and social practices.

Learning from each other is really important as a social practice. It is also important to know how to work together remotely. In global working environment, motivation is supported by assigning specific responsibilities to each role. Besides, AgileForUs has many principles to support the way of working together at a distance for global working environment.

AgileForUs recommends usage of specific frameworks and templates as a technical practice. Before starting AgileForUs process spiral, several *knowledge transfer* activities are held. In development cycle or maintenance

cycles, *practices* and *communication* are continuously done and communication framework is tended to be opened. Thus, multichannel communication is provided.

4. RESULTS

Results of this study are mainly derived from the interviews conducted with project managers at the companies and employee process assessment survey. Survey analysis is performed to evaluate the gap between the espoused way of working versus the way it is actually done at the project, what provides the ground for judging whether the right process is chosen for a particular project. Then, SFs, which have influence on software process selection, are determined and grouped into six main categories (Table 4). The most crucial SFs across the processes are highlighted to emphasize their importance for an organization. Further, analysis is extended with observations in what way these SFs have influence on selection process.

4.1 Process Assessment Survey

This section presents results of employee process assessment survey, conducted within both – ConsultIt and ItSystems projects.

According to Ericsson process assessment measurement mechanism, 4 metrics/key ratios were calculated and summarized in Table 3. The first “Processment” column of Table 3 is of most importance. It represents adoption rate for the particular process, indicating assimilation gap for the process – the difference between acquired and deployed practices of the process. We can see high adherence (more than 50%) for all the investigated processes, what indicates positive and quite high process adoption in the projects. Thus, we can use survey results as “evidence of match” for each particular project-process pair, i.e. that allows us to state that every examined project uses appropriate process. This fact, in its turn, gives us possibility to draw further conclusions based on environmental situation for each project.

	Processment	Process Commitment	Process Learning	Process Improvement
AgileForUs (ConsultIt)	100%	100%	100%	43,75%
SCRUM (ConsultIt)	66,6%	100%	100%	41,65%
Common Sense (ItSystems)	75%	100%	37,50%	43,75%

Table 3. Survey Analysis

4.2 Situational Factors Categories

The observations made herein are based on the interviews with project managers at the companies. The interview questions were asked to understand particular way of working and what PMs need throughout the project to get

the maximum advantage out of the process. The empirical data from these interviews together with literature review led us to determine the following 17 situational factors and group them into six main categories.

CATEGORY	SITUATIONAL / ENVIRONMENTAL FACTORS	AgileForUs	SCRUM	Common Sense
Team characteristics	Team size	Medium 10-15 people	Medium 5-10 people	Medium 5-10 people
	Working experience of team members	Medium It is more important to be skilled enough to perform tasks properly.	High Team members' specific competence , proactivity level and the ability of taking initiative is of big importance.	Medium Focus on individual skills and abilities.
	Education and awareness of team members on frameworks, tools and strategies	High Supported by trainings for all implementers in order to minimize issues regarding lost code and sticky errors.	Medium Supported by sprint based inspections, online presentations and separate meetings.	Medium Supported by department meetings, individual requests and assessments.
	Domain knowledge transfer	Medium Provides suggestions to how the domain knowledge can be spread e.g. knowledge transfer activities.	Medium	Medium
	Team Distribution	High It supports way of working for multi-site environment by usage of multichannel communication.	Medium	Medium
	Multicultural Team	High It promotes inspection to show onsite working way to offshore team.	Medium Quite Flexible - the person who is responsible for task puts rules to define the way of doing that specific task .	Medium
Project characteristics	Project Duration	Medium Flexible duration	Medium Flexible duration	Medium Flexible duration, most suitable for 1 year time period.
	Project Size	Suitable for large-scale projects as well as smaller projects.	Suitable for nearly any size of project.	Medium size of project.
	Acceptance of Requirement changes	High Project managers aim to have all possible changes during the project.	Medium Process itself is useful to handle customer change request on requirements.	High
	Quality level	High Focus on a good quality per iteration, so quality attributes are very important	Medium	Medium
Organization characteristics	Monitoring Process Procedure	Medium Supported by daily meetings and the usage of commit everyday principle.	Medium Supported by daily meetings.	Low Supported by weekly meetings.
	Personal effect on working practices	Medium PM can introduce any process change he feels is necessary to improve the implementation.	Medium PM can make prioritization on behalf of customer.	Medium
	Certification	Medium Mature-supported by having CMMI standard (level 3)	Medium Mature-supported by having CMMI standard (level 3)	N/A
	Effectiveness of Resource Usage	Medium It prevents having two people doing same task.	High One person should be able to perform different tasks. E.g.: developers also should be able to perform test.	High One person should be able to perform different tasks.
Business Domain		Medium PMs try to adjust process to the customers requirements.	N/A	Medium
Contract Type		Flexible-depends on customer needs.	N/A	Fixed Price
Customer/Stakeholder Involvement		High It is important to involve customers to add additional value to the process.	High It is supported by monthly meetings to make prioritization what should be done next.	High

Table 4. Situational Factors

In the *Team Characteristics* category there are six SFs: team size, working experience of team members, education and awareness of team members on frameworks, tools and strategies, domain knowledge transfer, team distribution, and multicultural team. It helps to determine how development is distributed between offshore and onshore teams, located in different geographical locations. In all interviews project managers proposed to consider offshore and onsite teams as one unit as a key to success in outsourcing. In this sense, communication between distributed multicultural team becomes essential part of working process. With respect to this category, AgileForUs emphasizes education and team members' awareness of frameworks, tools and strategies, and SCRUM considers working experience of team members as a key, while Common Sense relies on individual skills and abilities.

In the *Project Characteristics* category four factors were identified: project duration, project size, acceptance of

requirements changes, and quality level. Here, AgileForUs differs from other processes in focusing on a good quality level for each and every increment, and therefore the degree of quality that product needs plays a decisive role in process selection.

In *Organization Characteristics* category four factors were determined: monitoring process procedure, personal effect on working practices, certification, and effectiveness of resource usage. It focuses on procedures for resource and cost allocation for the project, as well as how project related activities are monitored and distributed within organization. Except for effectiveness of resource usage, there is not so much difference among examined processes. SCRUM project managers emphasize the importance of giving multiple roles to an individual within one project to use human resources efficiently and increase their competence.

Business Domain, - one of independent categories, deals mostly with capability of a process to meet requirements of different business domain in a market. In one of the informal discussions ConsultIt project manager said: “Project managers try to adjust AgileForUs to the customer requirements, so for 98% of the cases AgileForUs is applicable to use, and for the other 2% it can be used with more traceability support.”

Customer/Stakeholder Involvement is other independent category, which is important for selection process. AgileForUs considers customer/stakeholder involvement as a very important part of the process, and to satisfy all their needs and requirements. AgileForUs is extremely flexible in making changes at any phase. Furthermore, customer/stakeholder’s feedback is considered most valuable response to add additional value to the process.

On the other hand, one of ConsultIt project managers said during interview: “SCRUM itself is useful and sufficient to handle customer’s change request”. Moreover, customers are not allowed to interrupt or interfere in process in the middle of the sprint, and they are adapted quite well to it. Same situation is observed for Common Sense process.

4.3 Answer to Research Question

Each project has different set of most influential characteristics, outlining particular environment it is run in. Different processes can share various SFs, but the level of their influence is different. So the “Project – Optimal Process” is characterized by a set of pairs “SF - Value”.

So in order to identify optimal process, PM needs to define environment the project is going to be run in the coordinates of SFs listed in the table 4 and position it.

It is also important to determine, which SFs can affect the project success most, and process selection needs to be adjusted in compliance with these factors.

For AgileForUs process, mostly team characteristics should be considered. In case of having multicultural and distributed environment, it is good to apply AgileForUs. It supports project managers in coordinating team by providing multichannel communication and supporting inspection activities. This process is also appropriate, when high acceptance of requirement changes is needed. For SCRUM, working experience of team members and effectiveness of resource usage should be considered as major situational factors. For Common Sense process we point out high acceptance of requirement changes together with multi-task resource usage as most influential factors.

5. ANALYSIS AND DISCUSSION

In this section we analyze SFs, which have been identified during interviews and provide discussion on them, compare with research done before and list the possible future work as an evaluation.

5.1 Related Work

A lot of research has been done to find out the reasons for software development project’s failures. In addition, some factors, such as poor user input, stakeholder conflicts, vague requirements, poor cost and schedule estimation, skills that do not match the job, which require attention have been outlined to decrease risk of failures (Lorin, 2001). Bekkers et al. (2008) conducted workshop on software product management and presented 27 situational factors explaining their influence on method selection. These factors helped us to shape the ideas about possible situational factors in the area of project management and software process improvement, which is in a focus for our study. Bern et al. (2002) analyzed contextual factors affecting software development process. This paper drew our attention to the fact of importance factors’ effect to the software development process. However, there is limited research conducted about situational factors that may help organization to choose appropriate software development process among specific agile processes.

5.2 Interviews Results

To choose appropriate process, project managers need to have sufficient knowledge about development process followed within organization. Also it is essential for them to understand, which development process is more powerful in which situation. Moreover, project managers need to know their team’s capacity with regard to practices and organizational needs. At this point, necessity of measurement becomes crucial to understand and improve practice.

During the interviews, we realized that each development process provides different advantages throughout the project depending on different situational factors. For instance, while AgileForUs is more efficient to work with distributed team in multicultural environment, SCRUM is more efficient in the resource management within an organization. The important thing is to identify which factors are crucial to make software development process selection based on.

On the other hand, in some situations particular changes can be done to the processes to use it more effective. For example, although the optimal number of team members is between 10 and 15 in AgileForUs, it can be used efficiently for larger team by adding more micro iterations into the process. Therefore, it could be useful to choose the process, which is easy to modify to meet the environmental needs.

The most important factors, which should to be looked for with regard to AgileForUs process, are: project size, quality level and team distribution – physical and cultural. When a project team is distributed in different geographical locations, organization needs to choose a process, which supports various communication ways between team members. In this case, it is beneficial to use AgileForUs,

which has activities that offshore and onshore team can share with each other.

Working experience of team members, acceptance of requirements changes, effectiveness of resource usage should be considered while choosing SCRUM. Project managers have to be aware of team members' working experience, and if they are experienced enough, it is advantageous to use SCRUM. Process itself is good to handle requirement changes in a case of having customers who tend to change their requests frequently. By following SCRUM, an organization can use resources more efficiently.

Generally, the project-process pairs, which have been examined in organizations, are matched properly. For us it

was also of a great interest to investigate, which process could be appropriate in what situation among our 3 cases. Table 5 presents cross-check evaluation among processes, and may serve as a roadmap for software process selection. Grey color here denotes current process, used for a particular project. Plus sign "+" indicates positive answer to the question whether the selected process is suitable for a particular environment, while minus sign "-" shows the contrary. Number of signs indicates the weight or level of confidence. Thus, in a given situation only ConsultIt 2nd project could be "upgraded" from SCRUM to AgileForUs process. The brief reasons for our conclusions are given in quotes.

	ItSystems	ConsultIt 1	ConsultIt 2
AgileForUs	--- "too heavy"	+++++	++
Common Sense	++++	--- "too wild"	--- "too wild"
SCRUM	-- "too formal"	--- "too plain"	+++

Table 5. "Process-Environment" Cross Check Evaluation

5.3 Limitations and Future Work

Our case study covers only three cases and was performed at two specific companies- IT solutions providers, and therefore our results are not completely generalizable. Future work is to extend the work with multiple cases including other processes, not only agile. Furthermore, it would be wholesome to conduct process assessment survey within whole organization, having more respondents to validate the approach completely.

6. CONCLUSIONS

The overall aim of this research was to increase industry managers' knowledge in the process assessment and process selection areas. The specific research objectives were to see the assimilation gap for the processes, and provide SFs as a guide to choose optimal process for each particular project.

We have achieved these goals and answered the stated Research Question in the context of environmental factors an organization should look for while choosing among multiple agile processes.

Findings and analysis, presenting which process is applicable in which situation, as well as a roadmap for software process selection, are based on empirical data obtained from the interviews together with literature review results. We also outlined possible future work and potential problems in generalizing and extending our approach to be applicable to other processes.

7. ACKNOWLEDGEMENTS

We are grateful to many people for their help, both direct and indirect, in writing this thesis work.

First of all, we would like to thank Dr. Lars Pareto – our academic supervisor for the care and patience, with which he reviewed the early versions of this paper, for the critical approach, valuable advices and conversations, which clarified our thinking on this and other matters.

The authors are also grateful to the company supervisors for providing resources for the thesis project, professional collaboration, interesting ideas and comments.

We also express deep thanks to Håkan Neeman, Alexander Tereshkin and Kiryl Bletsko, who shared their ideas with us, gave their time and comments, and provided us with constant support throughout the whole work.

And finally, the authors thank all the project managers, we interviewed during this thesis work, as well as all the employees, who participated in the Process Assessment Survey for their time and cooperation.

8. REFERENCES

Argyris, C., & Schon, D. (1996). Organizational Learning II -Theory, Method and Practice. Reading, Massachusetts: Addison-Wesley.

Anderson K. & McAdam R. (2004). A critique of benchmarking and performance measurement. Lead or lag?

Benchmarking: An International Journal Journal, Vol. 11, No.5, pp. 465-483.

Baskerville, R., Mathiassen, L., & Pries-Heje, J. (2005). Agility in Fours: IT Diffusion, IT Infrastructures, IT Development, and Business. In R. Baskerville & L. Mathiassen & J. Pries-Heje & J. DeGross (Eds.), *Business Agility and Information Technology Diffusion* (pp. 3-10). New York: Springer.

Bekkers, W., Weerd, I.v., Brinkkemper, S., Mahieu A. (2008) The Influence of Situational Factors in Software Product Management: An Empirical Study. 2008 Second International Workshop on Software Product Management.

Bern, A., Nikula, U., Pasi, S. J. A., Smolander, K. Contextual Factors Affecting the Software Development Process – An Initial View.

Bogdan, R. C. & Biklen, S. K. (2006). *Qualitative research in education: An introduction to theory and methods*. Allyn & Bacon. ISBN 978-0205512256.

Borjesson, A. (2006a). Making Software Process Improvement Happen. Doctoral Thesis. (ISSN: 1652-490X;4).

Borjesson, A. (2006b). Improve by Improving Software Process Improvers. *International Journal of Business Information Systems*, 1(3), 310-338.

Borjesson, A. and Mathiassen, L., (2004). "Successful Process Implementation". *IEEE Software*, Vol. 21, No. 4, pp. 36-44.

Borjesson A., Baaz A., Pries-Heje J., and Timmeras M., (2007). "Measuring Process Innovations and Improvements", In IFIP International Federation for Information Processing, *Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda*, (Boston: Springer).

Cattaneo F., Fuggetta A., and Sciuto D. (2001). Pursuing Coherence in Software Process Assessment and Improvement, *Softw. Process Improve. Pract.*, Vol. 6, No.1, pp.3-22

Douglass, B., P. (1999). *ROPES: Rapid Object-Oriented Process for Embedded Systems*. I-Logix Inc.

El Emam, K. and Briand, L. (1997) Costs and benefits of software process improvement. In *Better Software Practice for Business Benefit: Principles and Experience*. IEEE Computer Society Press, Washington, DC.

Enskog, U., (2006). "Process Awareness and Use at Ericsson AB," B.Sc. Thesis, IT University of Goteborg, Sweden.

Fayad, M. E. and Laitinen, M. (1997) Process Assessment Considered Wasteful. *Communications of the ACM*. Vol. 40, No. 11, pp.125-128.

Fichman, R. & Kemerer, C. (1999). The Illusory Diffusion of Innovation: An Examination of Assimilation Gaps. *Information Systems Research*, 10(3), 255-275.

Galliers, R.D.(1992), Choosing an Information Systems Research Approach, in *Inf. Systems Research: Issues, Methods, and Practical Guidelines*, R.D. Galliers, Editor, Blackwell Sci. Publ.: Oxford, UK. p. 144-162.

Grady, R.B., (1992). "Practical Software Metrics for Project Management and Process Improvement", Upper Saddle River, New Jersey, Prentice Hall.

Grady, R. B., (1997). "Successful Software Process Improvement", Upper Saddle River, NJ: Prentice Hall.

Highsmith, J. (2002). *Agile Software Development Ecosystems*. Addison-Wesley Professional. ISBN-10: 0-201-76043-6.

Humphrey, W., (1989). "Managing the Software Process", Addison Wesley Professional.

Kotter, J. P. (2007). Leading Change: Why Transformation Efforts Fail. *Harvard Business Review*, 85(1), 96-103.

Lorin J. M., Major Causes of Software Project Failures. *Crosstalk: The Journal of Defense Software Engineering*.

Moore, G. (2002). *Crossing the Chasm: Marketing and Selling Technology Products to Mainstream Customers* (rev. ed.). New York: Harper Collins Publishers.

Neeman H., Sarlin, A. (2007). Introducing the Implementation Workshop to a Technology Supplier working with a customer. M.Sc. Thesis, IT University of Goteborg, Sweden.

Paulk, M.C., Weber, C.V., Curtis, B., and Chrissis, M.B., (1995). "The Capability Maturity Model: Guidelines for Improving the Software Process". Reading, Mass., Addison-Wesley Pub. Co.

Pries-Heje, J., & Tryde, S. (2001). Diffusion and adoption of IT products and processes in a Danish Bank. *Diffusing Software Product and Process Innovations* (M. Ardis, & B. Marcolin, Eds.), (pp. 17-34). Norwell: Kluwer Academic Publishers.

Rocha, D.J. (2008), Strengthening the validity of SPI measurements trough statistical analysis: a case study at Ericsson AB. B.Sc. Thesis, IT University of Goteborg, Sweden.

Rogers, E. M. (2003). *Diffusion of Innovations* (5th ed.). New York: Free Press.

Rose, J., Pedersen, K., Hosbond, J. H. and Kraemmergaard, P. (2007) Management competences, not tools and techniques: A grounded examination of software project management at WMdata, *Information and Software Technology*, 49, 605- 624.

Seidel, J. V. (1998). Qualitative Data Analysis, Qualis Research, <ftp://ftp.qualisresearch.com/pub/qda.pdf> .

SEMA, (2002). "Process Maturity Profile of the Software Community", Software Engineering Institute, Carnegie-Mellon University.

Soy, Susan K. (1997). The case study as a research method. Unpublished paper, University of Texas at Austin. <http://www.ischool.utexas.edu/~ssoy/usesusers/1391d1b.htm>.

Stake, R (2000) 'Case Studies'. In N.Denzin and Y.Lincoln (eds), Handbook of Qualitative Research, 2nd edn (pp. 435-54). Thousand Oaks, CA:Sage.

Susman, G., & Evered, R. (1978). An assessment of the scientific merits of action research. Administrative Science Quarterly, 23(4) 582-603.

Weerd, I. v., Brinkkemper, S., Nieuwenhuis, R., Versendaal, J., & Bijlsma, L. (2006). On the Creation of a Reference Framework for Software Product Management: Validation and Tool Support. Proceedings of the 1st International Workshop on Product Management, (pp. 3-12). Minneapolis/St. Paul, Minnesota, USA.

Weinberg, G.M. (1997). Quality Software Management: Volume4, Anticipating Change. New York, New York: Dorset House Publishing.

Yin, R.K. (1994). Case Study Research. Newbury Park, California: Sage Publication.

Zahran, S., (1997). "Software Process Improvement: Practical Guidelines for Business Success". Addison-Wesley.

Yin, R.K. (1994). Case Study Research. Newbury Park, California: Sage Publication.

Zahran, S., (1997). "Software Process Improvement: Practical Guidelines for Business Success". Addison-Wesley.

APPENDIX 1

Process Assessment Survey (Processment) ConsultIt AgileForUs Process

Case 1: ConsultIt AgileForUs Process									
The Metric Processment									
Q1:I am familiar with the process I am expected to follow									
Q2:I think I work according to the process I am expected to follow									
	Positive (4-6)		100%						
Q3:I think the process I am expected to follow supports me in my work									
	Negative (1-3)		0%						
Q4:I use the process I am expected to follow to get the process material I need in my work									
The Metric Process Commitment									
Q5:I think the process I am expected to follow is needed									
	Positive (4-6)	0%	100%	Q5	+	-			
expected to follow is needed									
	Negative (1-3)	0%	0%	Q6	4	0	100%	0%	
		Negative (1-3)	Positive (4-6)						
Q6:I feel that my organization promotes the use of the process I am expected to follow									
The Metric Process Learning									
Q7:I think I have good enough knowledge in the process to help my colleagues									
	Positive (4-6)	0%	100%	Q7	+	-			
	Negative (1-3)	0%	0%	Q8	4	0	100%	0%	
		Negative (1-3)	Positive (4-6)						
Q8:I have colleagues with good enough knowledge in the process to help me with solving problems									
The Metric Process Improvement									
Q9:I think my organization does process improvements where it is needed									
	Positive (4-6)	18,75%	43,75%	Q9	+	-			
	Negative (1-3)	6,25%	31,25%	Q10	3	1	75%	25%	
		Negative (1-3)	Positive (4-6)		4	0	100%	0%	
									100,00%
Q10:I think my organization does process improvements that are good									

APPENDIX 2

Process Assessment Survey (Processment) ConsultIt SCRUM Process

Case 2: ConsultIT SCRUM Process									
The Metric Processment									
Q1:I am familiar with the process I am expected to follow									
Q2:I think I work according to the process I am expected to follow	Positive (4-6)	66,6%							
Q3:I think the process I am expected to follow supports me in my work	Negative (1-3)	33,4%							
Q4:I use the process I am expected to follow to get the process material I need in my work									
The Metric Process Commitment									
Q5:I think the process I am expected to follow is needed	Positive (4-6)	0%	100%	Q5	+	-			
expected to follow is needed	Negative (1-3)	0%	0%	Q6	6	0	100%	0%	
		Negative (1-3)	Positive (4-6)						
Q6:I feel that my organization promotes the use of the process I am expected to follow									
The Metric Process Learning									
Q7:I think I have good enough knowledge in the process to help my colleagues	Positive (4-6)	0%	100%	Q7	+	-			
	Negative (1-3)	0%	0%	Q8	6	0	100%	0%	
		Negative (1-3)	Positive (4-6)						
Q8:I have colleagues with good enough knowledge in the process to help me with solving problems									
The Metric Process Improvement									
Q9:I think my organization does process improvements where it is needed	Positive (4-6)	25,00%	41,65%	Q9	+	-			
	Negative (1-3)	8,35%	25,00%	Q10	4,998	1,002	83,3%	16,7%	
		Negative (1-3)	Positive (4-6)						
					100,00%				
Q10:I think my organization does process improvements that are good									

APPENDIX 3

Process Assessment Survey (Processment) ItSystems Common Sense Process

Case 3: ItSystems Common Sense Process									
The Metric Processment									
Q1:I am familiar with the process I am expected to follow									
Q2:I think I work according to the process I am expected to follow	Positive (4-6)	75%							
Q3:I think the process I am expected to follow supports me in my work	Negative (1-3)	25%							
Q4:I use the process I am expected to follow to get the process material I need in my work									
The Metric Process Commitment									
Q5:I think the process I am expected to follow is needed									
expected to follow is needed	Positive (4-6)	0%	100%	Q5	+	8	0	100%	0%
	Negative (1-3)	0%	0%	Q6	-	8	0	100%	0%
Q6:I feel that my organization promotes the use of the process I am expected to follow									
The Metric Process Learning									
Q7:I think I have good enough knowledge in the process to help my colleagues									
	Positive (4-6)	37,50%	37,50%	Q7	+	8	0	100%	0%
	Negative (1-3)	12,50%	12,50%	Q8	-	4	4	50%	50%
Q8:I have colleagues with good enough knowledge in the process to help me with solving problems									
The Metric Process Improvement									
Q9:I think my organization does process improvements where it is needed									
	Positive (4-6)	18,75%	43,75%	Q9	+	6	2	75%	25%
	Negative (1-3)	6,25%	31,25%	Q10	-	8	0	100%	0%
Q10:I think my organization does process improvements that are good									

APPENDIX 4

AgileForUs Interview Questions

1. What is AgileForUs? How do you position AgileForUs?
2. What are the conceptual principles of AgileForUs, i.e. what is the model of AgileForUs?
3. In what is the power of Agile? and AgileForUs as a model actualizing Agile Global Excellence?
4. What do you use to measure project progress in AgileForUs?
5. Which features does AgileForUs have as alternative way of monitoring process?
6. Which activities do you have for maintaining software architecture? What are the best practices “on how to communicate design, test approaches
7. Do you use OOD (Object-oriented design)?
8. How much effort do you put on design while preparing first version?
“Designers are at this stage encouraged to produce a wide spectrum of models.” Could you specify these models?
9. Do you encourage pair programming?
10. Roles vs. Functional specialists (such as developers, testers, analysts and managers). Any specific roles in AgileForUs? Team roles and management roles.
11. Is there a difference between a leader and a manager within AgileForUs? Does/should the team understand the difference between a leader and a manager?
12. Does the AgileForUs training/coaching recognize cross-functional teams and coordination?
13. In which phases do you encourage use of modeling tool and why?
14. Can you give any example of technique or practice occurred in Process Improvement phase with regard to - "How to":
 - a. Improve the productivity of the development team
 - b. Improve the engineering practices and tools in order to make each increment of functionality potentially shippable
 - c. Improve the awareness of the team's progress
15. What are potential /existing impediments have you seen/faced implementing AgileForUs?

Interview Questions with Project Manager 1

1. What is your software development lifecycle? (Core activities, main principles, etc.)
2. Which activities need to be done iteratively? Do you have any iteration within phases?
3. Do you use continuous integration? Do you follow any strategy to avoid architectural rule violation?
4. How do you monitor the progress of the project?
5. How often do you have project meetings?
6. How do you ensure that new source code integrated into the repository correctly?
7. What do you use to measure project progress?
8. How do you preserve software architecture?(how to communicate design, test approaches and tools)
9. Do you use any modeling tool? For which phases and why do you use it?
10. How do you perform verification/testing?
11. Do you use automation for verification/testing?
12. Do you use automation tools in any other phases of process lifecycle?
13. Which roles are involved in the project? How do you split the roles in each phase?

14. Does each role have specific role definition and responsibilities?
15. Are customers and stakeholders involved in SDP? In which particular phase and in what way are they involved in the process?
16. Do you have distributed teams? Do you use outsourcing? Which part of project do you outsource?
17. Do you have any activities to make team members aware of tools-to-be-used, frameworks and strategy?
18. Is there a fixed list of deliverables (incl. documentation) per each phase?

Interview Questions with Project Manager 2

1. What is the size of your project and team?
2. Which software development process do you use? What are main activities and principles do you follow?
3. Why do you select the development process you are using and your way of working according to it?
4. Which activities need to be done iteratively? Do you have any iteration within phases?
5. How often do you make your releases?
6. Do you use continuous integration? Do you follow any strategy to avoid architectural rule violation?
7. How do you monitor the progress of the project:
8. How often do you have project meetings?
9. How do you ensure that new source code integrated into the repository correctly?
10. What do you use to measure project progress?
11. How do you preserve software architecture?(how to communicate design, test approaches and tools)
12. How much effort do you put on design while preparing first version?
13. Do you use any modeling tool? For which phases and why do you use it?
14. How do you perform verification/testing?
15. Do you use automation for verification/testing?
16. Do you use automation tools in any other phases of process lifecycle?
17. Which roles are involved in the project? How do you split the roles in each phase?
18. Does each role have specific role definition and responsibilities?
19. Are customers and stakeholders involved in SDP? In which particular phase and in what way are they involved in the process?
20. How clear are the requirement specifications, which are provided by the customer?
21. How often do customers want to change requirements or add new ones to the requirement specifications?
22. Do you have distributed teams? Do you use outsourcing? Which part of project do you outsource? Which difficulties do you face while working with distributed teams?
23. Do you have any activities to make team members aware of domain knowledge, tools-to-be-used, frameworks and strategy?
24. Is there a fixed list of deliverables (incl. documentation) per each phase? Do you use any reusable artifacts such as templates, codes, modules, etc that company provides?
25. Did you consider the level of experience of your team members while selecting them for this project?