Software Product Line Scoping and Requirements Engineering in a Small and Medium-Sized Enterprise: An Industrial Case Study

Ivonei Freitas da Silva, Paulo Anselmo da Mota Silveira Neto, Pádraig O’Leary, Eduardo Santana de Almeida, Silvio Romero de Lemos Meira

1Federal University of Pernambuco (UFPE) – Recife, Pernambuco, Brazil
2Federal University of Bahia (UFBA) – Salvador, Bahia, Brazil
3Reuse in Software Engineering (RiSE)
4Lero - the Irish Software Engineering Research Centre

{ifs3, srlm}@cin.ufpe.br {esa}@dcc.ufba.br
{pamsn}@rise.com.br {padraigoleary}@gmail.com

Abstract

Software Product Line (SPL) engineering has been applied in several domains, especially in large-scale software development. Given the benefits experienced and reported, SPL engineering has increasingly garnered interest from small to medium-sized companies. It is possible to find a wide range of studies reporting on the challenges of running a SPL project in large companies. However, very little reports exist that consider the situation for small to medium-sized enterprises and these studies try develop universal truths for SPL without lessons learned from empirical evidence need to be contextualized. This study is a step towards bridging this gap in contextual evidence by characterizing the weaknesses discovered in the Scoping (SC) and Requirements (RE) disciplines of SPL. Moreover, in this study we conducted a case study in a Small to Medium sized Enterprises (SMEs) to justify the use of agile methods when introducing the SPL SC and RE disciplines through the characterization of their bottlenecks. The results of the characterization indicated that ineffective communication and collaboration, long iteration cycles, and the absence of adaptability and flexibility can increase the effort and reduce motivation during project development. These issues can be mitigated by agile methods.
Keywords:
Requirements Engineering, Scoping, Agile Methods, Software Product Line, Case Study.

1. Introduction

Software Product Lines (SPL) has been applied in a wide variety of domains, such as driver firmware (Iwasaki et al., 2010), security inspection (Li, 2010), enterprise resource planning (Hamza et al., 2010), and operational research (Demir et al., 2010), as a means to achieve quality improvements, and reductions in time to market (Pohl et al., 2005) (Clements and McGregor, 2012). Despite the aforementioned benefits, during SPL adoption several issues must be overcome by the company such as the required up-front investment to achieve an operational SPL, challenging adoption barriers, and the lack of guiding empirical studies in the field (Bastos et al., 2011).

The scoping (SC) and requirements (RE) disciplines are extremely important to a SPL, as they define the initial activities and steps of the SPL life-cycle. For this study, we consider SC as a planning activity that defines the boundaries of the SPL by deciding which features are “in” (economically relevant to be included as SPL core assets) and “out” (not economically relevant). We consider RE as the statements that describe the features such as the behavior descriptions, properties, qualities, and the constraints that the feature should satisfy. To specify these statements, we used textual specifications for features, requirements, and use cases. Thus, a feature can represent a requirement, a selection amongst optional or alternative requirements, nonfunctional requirements, and implementation characteristics. A set of features describes a domain.

There are few industrial studies in the literature that characterize the SPL SC and RE disciplines. Moreover, studies that justify the use of agile methods to improve the SPL SC and RE disciplines are scarcer. All these studies try develop universal truths for SPL but the lessons learned from empirical evidence need to be contextualized (Dyba, 2013).

The goal of this study is to justify the use of agile methods when introducing the SPL SC and RE disciplines in a SME through the characterization of their bottlenecks with contextualized empirical evidence (Dyba, 2013). We grouped characterized bottlenecks and identified the weaknesses in a number of aspects that the literature mentions as success factors for agile software
development (Silva et al., 2011). The success factors are: the effort (Dybå, 2000; Hazzan and Hadar, 2008; John and Eisenbarth, 2009), communication and collaboration (Dybå, 2000; Hazzan and Hadar, 2008; Pettersson et al., 2008; Stelzer and Mellis, 1999; Niazi et al., 2005), iteration and adaptation (Hazzan and Hadar, 2008; Pettersson et al., 2008), motivation (John and Eisenbarth, 2009), and requirements and technology volatility.

These factors were chosen, mainly, because of a systematic mapping study into SPL and agile methods (Silva et al., 2011). In this mapping study, the primary studies partially addressed each one of these factors.

As these factors can have different meanings, for this study we defined them as:

- **Effort factor.** The time spent by one participant on development tasks such as specify features and validate requirements.

- **Communication and collaboration factor.** The interactions among the team members and how these interactions contribute to the development of the tasks.

- **Motivation factor.** The participant’s feelings regarding the tasks associated with a discipline before, during, and after performing them.

- **Iterativeness factor.** The potential of the disciplines to foster the building scope and requirements artifacts through several iterations in sequence. In that each iteration is a self-contained mini-scoping or mini-requirements discipline composed of tasks such as identify features, specify requirements, and inspect use cases.

- **Adaptability factor.** The potential of the disciplines to foster adjustments in the artifacts, team, technology, and process to become more effective.

- **Volatility factor.** The changes from the customer needs, technology, and domain, which must be considered in the adopted process.

This paper discusses the challenges that emerged from the weaknesses and the lessons learned. The challenges are discussed from the point of view of mitigation strategies and the use of agile approaches to overcome identified bottlenecks.
The remainder of this paper is structured as follows: Section 2 describes related work. Section 3 describes the context in which we performed the case study. Section 4 discusses the study design, by stating the research questions, and the data collection, analysis and validity procedures. The results are presented in Section 5. Section 6 presents the threats to validity. Section 7 describes the main findings, the lessons learned, and identified challenges, and finally, Section 8 describes the conclusions, implications, and future work.

2. Related Work

Some studies have provided data about effort, use of communication, iterativeness, adaptation, motivation, or volatility regarding SPL SC or RE. The following studies are briefly described, since they provide empirical data regarding SPL SC or RE.

In the study by Knauber et al. (2000), the authors describe initial results and lessons the application of the PuLSE approach (Bayer et al., 1999) in six SMEs. Despite the limited resources in the companies, the scoping discipline contributed to the creation of a business vision and the identification of new business opportunities. However, the study provided few details regarding the effort, communication and collaboration, iterativeness, adaptability, motivation, and volatility.

In Gacek et al. (2001) and Verlage and Kiesgen (2005), the authors describe lessons learnt and drawbacks regarding the introduction of a SPL (using the PuLSE approach (Bayer et al., 1999)) in the company Software AG. The company transitioned from legacy systems to save development effort and get started on a stable platform of domain functionalities. The scoping team (from development, management, sales, and marketing units) was not a permanent team and the meetings occurred only when major scoping activities were required. Communication was considered effective and fast. The initial findings of these two studies also contributed to increase the body of evidence in the SPL area. However, they did not address weaknesses in iterativeness and motivation.

Complementing the discussion in Bayer et al. (1999), the work of Schmid (Schmid, 2002) presents a well-documented approach for SC and its extensive validation in Software AG and Bosch companies. The approach describes the product line, its domains and features and performs an assessment of the reuse benefits and risks, while identifying assets for the product line. The
extensive validation was accomplished in the case studies. The results have contributed to SPL scoping. However, the study focused only on the effort factor.

Herrmann and Liebehenschel discuss their experiences in a study (Herrmann and Liebehenschel, 2011) focusing on effort, communication, feedback, adaptability, and volatility when performing requirements engineering for SPL. The study applied SPL RE in several automotive systems. Although they present various aspects of SPL RE that helped them to generate the SPL requirements documents, detailed information about what they consider scoping as well as information on weaknesses in effort, communication, iterativeness, adaptability, motivation, and volatility are missing.

Yu, Geng, and Wu discuss a case study (Yu et al., 2012) that evaluated an approach to provide traceability between requirements and features for individual applications within same domain. After the approach defines a feature tree and establishes traceability with requirements for each application, the approach then merges all the models to form a domain feature tree model as well as traceability between the domain features and requirements. Although the study has relevant aspects, such as the traceability between features and requirements, the paper does not describe the factors addressed in our work.

Noor describes two studies (Noor et al., 2006, 2008), where the collaboration factor is alleviated through the collaboration engineering technique. This improves communication and collaboration between SPL stakeholders during the SPL SC. The authors organized the approach in three different layers: (i) process layer, which defines the objectives, tasks and participants of the process, (ii) pattern layer, where the process is modeled using patterns from collaboration engineering, and (iii) thinKLets layer, where the tasks are decomposed to allow their execution using thinKLets.

The approach facilitates stakeholder involvement and the results are based on an industrial context with the reengineering of legacy systems into a SPL. However, results about the effort, motivation, and volatility variables were not reported in the studies. The aforementioned studies partially address our objective. As stated previously in this section, the studies focused on one discipline, for example scoping, or on a few variables, for example, effort and collaboration. They did not aim to investigate, qualitatively, the weaknesses of several variables for scoping and requirements disciplines as well as their effects on each other. This paper investigates the effort, communication and collaboration, iterativeness, adaptability, and volatility during the SPL SC
and RE disciplines through a qualitative study. As a result, this study aims to justify the use of agile methods when introducing the SPL SC and RE disciplines. Additional information about these studies and others that addressed SPL SC or RE, although with limited data about our studied aspects, can be found in John and Eisenbarth (2009), where the authors conducted a survey on approaches for the scoping discipline, and in Alves et al. (2010), which reports a systematic review on the requirements discipline in SPL.

3. Case Study Context

Considering that the context can cause misinterpretations on what SMEs are, and make more difficult to generalize the answers, challenges, and lessons learned of the study, we show real information about the company and context in the following section.

3.1. The company

We carried out the case study within a Brazilian software development company which develops systems in the domain of information systems for medical management, since the year 1994. It was created offering strategic and operational integrated solutions for hospitals, clinics, labs, and private offices. The company currently has more than fifty clients across many states in Brazil.

The company has about fifty employees (about six developers, an architect, and requirement analysts) and maintains four products presented in decreasing order according to their sizes. Product A has features which belong to thirty-five sub-domains and manages all sections of a hospital, from financial to patient aspects. Product B has features which address twenty-eight sub-domains, including clinical management supporting activities related to medical exams, diagnostics, and other related tasks. Product C has features which belong to twenty-eight sub-domains to manage labs of clinical pathology. Finally, Product D is a web product which addresses eleven sub-domains to manage the tasks and routines of a private office.

Figure 1 shows a partial feature diagram for the products. The complete domain has about six million lines of source code distributed across 373 files, and the company is operating in this domain for fifteen years.

A previous investigation identified that the company development and maintenance teams are conceptually divided along with two solutions, the
desktop product and web-based units (Bastos, 2011). Even with such a division, it is common to exchange activities among employees, and furthermore, some of them hold more than one role in the same project.

The staff presented deficiencies in the application of important software engineering practices such as requirements analysis and design, configuration management, and testing. Additionally, the company does not meet the goals for the process areas in CMMi (CMMI, 2010), for this reason it is not possible to associate the company to any maturity level. The existing documentation is composed of a wiki system with a description of the domains, training manuals, and single-system screen images.

Within this context and through partnership with RiSE Labs\(^1\), the company decided to introduce SC and RE. During their execution, experts in the domain, business, and legacy products (from company) validated the artifacts produced by a SPL team. In total, a staff with twelve members (from company and RiSE) was involved in the SPL project.

### 3.2. Object of Study

The SC and RE disciplines are the object of study in this paper. They are components of the SPL approach named RiPLE (RiSE Product Line

\(^1\)labs.rise.com.br
The approach also includes the design (Filho et al., 2009), test (Machado et al., 2011), evolution (Oliveira, 2009), and risk management (Lobato et al., 2012b) disciplines. If we compare the SEI framework (Northrop and et. al., 2007) with RiPLE, the former is more comprehensive as it addresses organizational and technical management beyond the common software engineering activities. However, the RiPLE is a process with work product, roles, and tasks to be followed by the practitioners. The SC discipline consists of four main phases (Balbino et al., 2011): Pre-Scoping, Domain Scoping, Product Scoping, and Assets Scoping. Pre-scoping aims to identify relevant characteristics that will influence the next stages of the scoping process, such as business goals, team profile, operational and organizational contexts, and market aspects. Moreover, this phase aims to promote initial contact between the project team and customers. Domain scoping has the objective to systematically analyze and discuss the domains among the project members. Product scoping aims to identify the particular products that will be developed and the features that they should provide. Assets scoping aims to identify the assets which should be developed in a reusable way. In this phase, the scope is aligned with the business goal previously determined by the stakeholders during the pre-scoping phase.

The RE activities are responsible for refining the scope. The scope definition is a pre-requisite to starting the process which has the product map artifact as a mandatory input. The RE encompasses three main steps (Neiva et al., 2010): model scope, define requirements, and define use cases. The objective of the model scope step is to specify and describe features, which composes the core asset platform. The define requirements step aims to specify and describe the requirements associated with the features previously described in the functional view. The objective of the define use cases step is to specify and describe the use cases associated with the requirements previously described in the process view. These are the steps required to perform a specific operation.

The RiSE Labs developed and we adopted a tool (Cavalcanti et al., 2011) to support the features, requirements, and use case specification. In addition, we applied the inspection activity to verify and validate the artifacts generated in both disciplines (Souza et al., 2013).
4. Case Study Design

The research method applied in this study was an embedded, flexible, exploratory, and single case study (Runeson and Höst, 2009; Yin, 2003). Embedded because the case study encompassed two analysis units (the scoping and requirements disciplines). Flexible as new information during data collection can be important or critical for the study, and this way, the study design can be updated based on this feedback. Exploratory since we are interested in understanding the SC and RE disciplines to explore the nature and weaknesses associated with them within a certain context (Patton, 2001) (a software industry). Finally, it is single-case study since the company develops several products with similarities between them in the same domain.

This case study is based on the general process (see Figure 2) defined in Runeson and Höst (2009). The first step is the case study design that defines the goals and the planning of the study. The second step, preparation for data collection, defines the procedures and protocol of the study. The third step, collecting evidence, is the execution of the data collection procedure. The fourth step, analysis of collected data, is the analysis procedure of the data, describing the conclusions. Finally, the fifth step, reporting, packages the study to be reported.

![Figure 2: Case study general process.](image)

4.1. Research Questions

As previously mentioned, the main objective of this study is to justify the use of agile methods in the SPL SC and RE by characterizing them in terms of possible weaknesses. In order to address this objective, the effort, communication and collaboration among the stakeholders, iterativeness and adaptability of the process, motivation, and requirements and technology volatility were the variables observed. Thus, in order to characterize these issues we defined the following research questions:
• How do the stakeholders characterize the effort to perform SPL SC and RE? Rationale: This question aims to identify the main bottlenecks regarding effort in terms of man-hours when performing the scoping and requirements disciplines. Besides the effort distribution amongst the disciplines, the tasks are characterized as well. As a result, the answer for this question can serve as input to the definition and establishment of agile practices to overcome this issue.

• How do the stakeholders characterize communication and collaboration in SPL SC and RE? Rationale: Characterizing how communication and the collaboration are performed by the stakeholders can aid in understanding the inception of the scoping and requirements issues, and to identify opportunities to apply agile practices that overcome these issues.

• How do the stakeholders characterize the SPL SC and RE process iterativeness and adaptability? Rationale: Interactivity and adaptability are important aspects for improvement, feedback, and reflection of the process. Both SPL and agile processes must be adjustable to the company context. A process that is not adjustable can be frustrating and demotivating for the stakeholders. Thus, it is important to identify where, when, and how we can make the process more adequate to the company and their developers.

• How do the stakeholders characterize the motivation with SPL SC and RE? Rationale: Different factors may contribute to the motivation. In this study, we would like to understand whether different variables, such as communication, organization, documentation, effort, or iteration might impact on the motivation within the project environment.

• How do the stakeholders characterize SPL SC and RE to deal with volatile requirements and technology? Rationale: SPL works well with stable domains. However, frequent changes in domain, requirements, or technology encourage the application of agile approaches. Considering the company context, Section 3, it has migrated its systems slowly from desktop to web platform. Functionalities have been discarded according to domain analysts from the company.
4.2. Case and Participants Selection

The basic conditions to address our problem, goal, and research questions are: any SME that has a legacy single-system portfolio in a specific domain; their products share commonalities; and the company is involved with the study.

After an initial search in the Brazilian company catalog and contacts, the RiSE Labs (henceforth named as SPL team) established a collaborative project with a company (henceforth named as the company) that works in the domain of information systems for medical management.

Twelve participants were selected based on the convenience sampling method (Wohlin et al., 2000). This selection considered different roles and profiles involved in SC and RE disciplines performed within the company. For the scoping discipline, we selected a domain expert (from the company), a scoping expert (from the SPL team), a domain analyst (company), an architect (company), developers (SPL team), a risk manager (SPL team), and a product line manager (SPL team). For the requirements engineering discipline, we selected two requirements analysts (SPL team), a risk manager (SPL team), and a requirements inspector (SPL team).

As our case is a SME, we assumed that a role could be played by more than one engineer, i.e., the domain expert and architect roles could be performed by the same participant. In such cases, the communication and collaboration variable could not be captured.

4.3. Data Collection Procedures

This study adopted four data collection methods, namely documentation analysis, observation, focus group, and interviews.

**Documentation Analysis.** The analysis of the documents is a technique which focuses on the documentation generated by software engineers (Seaman et al., 2007). All documents related to SPL SC and RE disciplines were analyzed including the reports from the project management activity, as well as the features, requirements, and use cases specifications. We analyzed other informal documentation artifacts: a wiki system, product screenshots, and messages (by e-mail) from stakeholders.

In this method, the effort, communication and collaboration, iterativeness, adaptability, and volatility variables could be captured.
Effort data was recorded in the project management documents. The activities, dates, time, and involved roles are the most common types of recorded data.

Communication variables could be found, e.g., in the messages exchanged between the stakeholders, and meeting proceedings. Sent messages and the response time for them are examples of relevant data that characterize this variable.

The records of the adopted version control system, exchanged messages between the stakeholders, and meeting proceedings have evidence about the iterativeness and adaptability variables. Data such as the number of changes, iterations between disciplines, and adjusts in the team or process, aided to characterize these variables.

Observation. This method followed the advices from Seaman (Seaman et al., 2007), in specific, the use of the following methods: think aloud and direct observations, field notes with place, time, participants, tone and mood of the interactions, and observer’s comments.

The observed roles were: scoping experts, developers, SPL management, architects, domain analyst, and requirements analysts and inspector. Both disciplines were observed over 120 hours. This was broken down into 40 hours in scoping and 80 hours in requirements engineering.

We captured the effort by observing the time required to accomplish the task by one person, the communication and collaboration by observing the interactions between stakeholders when exchanging messages either by email or chat, and in face-to-face interactions during meetings. We also captured requirements volatility when interacting with the domain expert and domain analyst.

As SC and RE disciplines do not have iterations between the disciplines, we could not characterize such a variable.

We observed the adaptability when inspecting some artifacts (mainly features and requirements). In this case, after the inspection (Souza et al., 2013), the suggested changes were incorporated into the templates to specify features, requirements, and use cases.

Focus Group. We performed focus group meeting, as advocated in Kontio (Kontio et al., 2007). These meetings took 34 minutes in which time we discussed the improvements to the SPL SC and RE disciplines in terms of challenges and lessons learned, and their relationship with agile software development. We selected the main developers (SC), requirements analysts and inspectors (RE), and risk manager (SC and RE) to participate in the
focus group, since they actively interacted across both disciplines.

We characterized all the variables by analyzing the answers to the open questions from participants. Although the motivation variable could be captured in variables, some additional reasons that could affect this variable were identified.

**Interviews.** Face-to-face interviews were conducted after finishing the SC and RE disciplines. We identified most weaknesses in the SC and RE disciplines with this method. The participants were free to talk about the process deficiencies, difficulties and benefits of the project. Similarly to the focus group method, all the variables could be captured during the interview, although at this point we only analyzed the answers for the open-ended questions.

4.4. Analysis Procedure

![Figure 3: Approach for data collection and analysis.](image)

Based on Miles and Huberman (*Miles and Huberman, 1994*), this study analyzed the collected data qualitatively. Figure 3 shows the overall analysis procedure.
After data collection, we reduced the body of data to a comprehensible format by coding based on the study goals. A code list\(^2\) (see Section 5, Table 2) was categorized considering the interest of the research questions. We identified some codes in the documentation, the observation notes, and the excerpts from the interviews and focus group audio. Each code represents events, variables, states, and so on that addresses directly or indirectly the research variables. For each code, we attached a memo\(^3\) with some proprieties or descriptions.

All of these codes described incidents, categories, and events occurring during the SPL SC and RE disciplines. After identifying them, we clustered and coded them in patterns to identify insights regarding the interactions in both disciplines.

We built the memos and drafted a causal loop diagram to facilitate the visualization of the code interconnections and causal-effect relationships.

The causal loop diagram (Sterman, 2000) or diagram of effect (Weinberg, 1992) focuses on the relationships and impacts that a specific code (variable) has on another code. A causal loop diagram can be used to explore which variables impact on others, i.e., what would be the effect on a variable or the project when code properties change (see Section 5, Figure 5).

These relationships can be a causal link (arrows), where an element can have an effect on another. The “O” symbol on the arrow means that an element has an opposite effect. For example, if an element increases, the other element decreases to a certain extent. The symbol “C” means a constraint on one element on another one. For example, “cash supply” element restricts the element “number of developers”.

Thus, whether the arrow has a direction from one variable (element) to another one, this means that the former variable has an impact on the latter. The opposite effect indicates that a former variable is impacted inversely. When the symbol “O” is not present, the effect is not inverted. The constraint symbol “C” represents the former variable constrains, e.g., resources on the latter one.

We wrote-up memos while we collected, coded, drafted the data, or drew

---

\(^2\)A list of phenomenons of interest to the researcher, which abstracts an event, object, action, or interaction that has a meaning for the researcher (Glaser and Strauss, 1967).

\(^3\)A memo can be a sentence, a paragraph or a few pages for example. Memos are used to write-up ideas about codes and their relationships. It is a theorized idea about codes (Strauss and Corbin, 1990).
the causal loop diagram. When drawing the causal loop diagram, we revised
the memos to comply with the diagram and vice-versa. The memos in Section
5, Table 4 are related to the causal links showing the main causal-effect
relations observed in the scoping and requirements disciplines.

The conclusion task in Figure 3 is responsible for deciding the data mean-
ing through the patterns, explanations, causal flows and propositions visu-
alized in the causal loop diagram. The patterns or themes, contrasts, com-
parisons, clustering, and counting are performed on the causal loop diagram
and the memos attached to the map are used to draw the conclusions.

Finally, discussion with participants of the project were performed in or-
der to test, verify, and validate the conclusions. For each data collection
instrument, these steps were considered inorder to make draft characteri-
izations on our insights, and suggestions regarding each research question.
Next, the conclusions gathered from all data collection instruments are used
to answer the research questions.

5. Results

In this section, the findings of the case study are presented describing
the SPL SC and RE disciplines performed within the company, through the
answers of the aforementioned research questions. Some general data from
the case study are depicted in Table 1.

5.1. Case Study Historic

<table>
<thead>
<tr>
<th>Period</th>
<th>SPL Discipline</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>January - May</td>
<td>Scoping</td>
<td>740h 58min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>840 identified features</td>
</tr>
<tr>
<td></td>
<td></td>
<td>478 hours</td>
</tr>
<tr>
<td>June - December</td>
<td>Requirements engineering</td>
<td>92 specified features</td>
</tr>
<tr>
<td></td>
<td></td>
<td>144 requirements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>327 use cases</td>
</tr>
</tbody>
</table>

Table 1: Analyzed SPL project disciplines.

The SPL SC discipline started in January, with the pre-scoping phase, followed by domain scoping, product scoping, and assets scoping.

During the pre-scoping, some meetings were scheduled with different project stakeholders to evaluate the SPL availability, benefits, drawbacks, and the domain in which it would be implemented.
Next, the domain scoping was held aiming to identify the domains and sub-domains with more potential to compose the products of the product line. The product scoping was performed to identify and review features, identify products, and construct and validate the product map. Finally, asset scoping created metrics, applied them, and prioritized the features on the product map. The timeline for each discipline is shown in Figure 4.

The SPL SC discipline was performed by nine people from the SPL team and three software engineers from the company, composed of one scoping expert, one product line manager, seven developers, one architect, one project manager, one market analyst, and one business analyst. One employee fulfilled the roles of project manager, domain expert, and architect. During the execution, we identified 3644 features. However, as the granularity was too fine-grained, the project members decided to adequate the features granularity according to Kang et al. (1990), contributing to increase the time spent during the feature identification. At the end, a pool of 840 features, four products, and 102 sub-domains were consolidated and validated during meetings with the SPL team members.

The wiki system and the products from the company were used to collect
data about the products, domains, and features. All of them were consolidated in a SPL vision document and stored in a repository.

The SPL RE discipline started with the model scope, followed by define requirements, and define use cases steps.

This case study focused on two iterations in the RE discipline. The first iteration was conducted by the requirements analysts and inspected by a different team. Each phase (model scope, define requirements, and define use cases) in the requirements engineering discipline resulted in the artifacts - features, requirements, and use cases.

After the requirements analysts performed all the phases, the inspection process was started to ensure the quality of the artifacts. The second iteration in SPL RE was performed with differences. For example, the inspection was carried out after each phase, that is, after the features were defined by the model scope step, the inspectors conducted meetings to inspect the specified features before starting the next step (define requirements). The timeline for each step and the inspection process are shown in Figure 4.

We collected data from the wiki system documentation, produced scoping artifacts, domain experts, and the legacy products.

Besides the inspection process, risk management activities were carried out to capture the challenges and risks emerged during the project (Lobato et al., 2012a,c).

Organizing its legacy products as a SPL and identifying which sub-domains should be developed as reusable assets were the main results for the company.

5.2. Codes, Clusters, and Memos of the Evidence

The code list, clusters, and memos aided understanding of how each research variable (effort, communication and collaboration, iterativeness, adaptability, motivation, and volatility) impacted on the others.

We consolidated the first code list in Table 2. There are several codes that are not defined as research variables for this study, however they still impacted on these. Moreover, all the codes were present in the captured evidence from the observations, interviews, and documents. Many of these codes were suppressed when the causal loop diagram (Figure 5) was drawn, in order to make the diagram more concise. The column Description explains or complements the meaning for each code.

<table>
<thead>
<tr>
<th>Code Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td></td>
</tr>
</tbody>
</table>
## Table 2 – continued from previous page

<table>
<thead>
<tr>
<th>Code Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Effort</strong></td>
<td>Stakeholders took 20 minutes to describe a use case, thus, many features implied in a big effort. Is the scalability impaired?</td>
</tr>
<tr>
<td><strong>Poor documentation</strong></td>
<td>Poor legacy documentation before starting the SPL project. The wiki system was the only legacy documentation describing the legacy products. SPL team did not pay attention to the wiki system because the products were the main way to capture and specify features, requirements, and use cases. New scenarios had no documentation.</td>
</tr>
<tr>
<td><strong>Incorrect description of features</strong></td>
<td>The inexperience of the SPL team with the domain contributed to misunderstanding between feature and requirements.</td>
</tr>
<tr>
<td><strong>Absence of the domain expert</strong></td>
<td>Domain experts were too busy with other company activities to remain onsite.</td>
</tr>
<tr>
<td><strong>Simulate operation system</strong></td>
<td>In order to identify features and use cases, the scoping expert simulated some scenarios in the legacy systems. This action identified many obsolete features. It was the main way to capture features, requirements, and use cases.</td>
</tr>
<tr>
<td><strong>Incorrect granularity</strong></td>
<td>The simulation of scenarios in the legacy systems to identify features generated many detailed features.</td>
</tr>
<tr>
<td><strong>Prioritize features, requirements, and use cases</strong></td>
<td>The prioritization of sub-domains was demotivating because there were much features, requirements, and use cases for each sub-domain, thus the iterations ended up being large.</td>
</tr>
<tr>
<td><strong>Poor Communication</strong></td>
<td>The company members had little commitment with the SPL project. The work was not integrated between the company team and SPL team, interaction face-to-face with the domain expert was not effective.</td>
</tr>
<tr>
<td><strong>Poor Collaboration</strong></td>
<td>Company team did not write features, requirements, or use cases together with the SPL team.</td>
</tr>
<tr>
<td><strong>Meetings</strong></td>
<td>The validations of the artifacts were performed during the meetings.</td>
</tr>
<tr>
<td><strong>Validation</strong></td>
<td>The SPL and company teams validated the domain, features, requirements, and use cases.</td>
</tr>
<tr>
<td><strong>Email and chat</strong></td>
<td>Whenever the domain experts were not present in the meetings, they sent their validation by email.</td>
</tr>
<tr>
<td><strong>Face-to-face</strong></td>
<td>Some questions were solved face-to-face, but the domain expert was too busy.</td>
</tr>
<tr>
<td><strong>Motivation</strong></td>
<td>Building the core assets incrementally, training in the products, and tools to manage features, requirements, and use cases motivated the SPL team. However, the participants stated as having little motivation due to the lack of interaction with other disciplines. The building of a SPL incrementally, including other disciplines, could increase the motivation in the SC and RE disciplines.</td>
</tr>
<tr>
<td><strong>Iterativeness</strong></td>
<td>Iteration time box was too long.</td>
</tr>
<tr>
<td><strong>Adaptability</strong></td>
<td>No activity was estimated. There was no adaptability activities. In the requirements discipline, some adaptations emerged because of the inspection activity.</td>
</tr>
<tr>
<td><strong>Reflection ad-hoc</strong></td>
<td>The reflection on the process was documented as lessons learned.</td>
</tr>
<tr>
<td><strong>Other disciplines</strong></td>
<td>The participants encouraged the interaction with other disciplines such as design, testing, and implementation.</td>
</tr>
<tr>
<td><strong>Time box</strong></td>
<td>The iteration time box took months. The participants considered it as ‘too long’. It should take few weeks.</td>
</tr>
<tr>
<td><strong>Inspection</strong></td>
<td>The inspection provided a rigor to specify the artifacts through the inspection patterns. Inspection built a template providing improvements in the specification.</td>
</tr>
<tr>
<td><strong>Volatile requirements and technology</strong></td>
<td>The SPL team did not contact customers (hospitals, clinics, labs, doctor office). However, the SPL team noticed changes in the requirements during the validation meetings with the domain experts.</td>
</tr>
</tbody>
</table>
Table 2 – continued from previous page

<table>
<thead>
<tr>
<th>Code Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System expert</td>
<td>SPL team suggested a new role for the project, named of legacy system expert.</td>
</tr>
<tr>
<td>Return value</td>
<td>The SPL team thought that the value of the artifacts is small to the company.</td>
</tr>
<tr>
<td>Scoping</td>
<td>The participants applied a formalized process for scoping defined in a master dissertation.</td>
</tr>
<tr>
<td>Decisions about features,</td>
<td>Several decisions were taken by email or chat, regarding the absence or</td>
</tr>
<tr>
<td>requirements, and use cases</td>
<td>presence of features, requirements, or use cases in the legacy products.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Requirements discipline took 2 iterations. There were ambiguous between</td>
</tr>
<tr>
<td></td>
<td>features and requirements. The company did not validate the requirements</td>
</tr>
<tr>
<td></td>
<td>artifacts.</td>
</tr>
</tbody>
</table>

Table 2: First generated code list with some comments.

As an example for code clusters, we grouped the codes *face-to-face* and *absence of the domain expert* into the group *communication*, since both address aspects about the communication between the company and the SPL team. Thus, communication becomes the code drawn in the causal loop diagram.

From the codes we derived twelve variables of the causal loop diagram: *effort, communication and collaboration, iteration and adaptation, motivation, requirements and technologies volatility, new role - product expert, and inspect meetings* variables. For example, it is intuitive to derive *inspect meetings* from the code *inspection*. However, we derived *company team availability* from the codes *absence of the domain expert, face-to-face, and simulate operation system*.

The mapping among the codes and variables is organized in Table 3. Although there are many other possibilities for the relationships among the code list and causal loop diagram variables, this mapping reflects the main insights from the participants when performing the SPL project.

The codes *scoping and requirements* were not mapped in the causal loop diagram because they represent all the variables.

With the code list and clustering performed (causal loop diagram), we built the memos. In Table 4, we show the memos regarding the impacts of a code on another one. The column *Description* explains events that justify that impact (causal-effect relationship).
Table 3: Mapping between code list and causal loop diagram variables

<table>
<thead>
<tr>
<th>Memo Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Iteration and adaptation to Effort</strong></td>
<td>There is evidence that the itterativenss and adaptability between the requirements discipline and inspection activity decreased the effort. The evidence is based on the fact that the requirements analysts directed the standardization of features, requirements, and use cases artifacts. In addition to decreasing their redundancy and inconsistency through inspection meetings.</td>
</tr>
<tr>
<td><strong>Communication and Collaboration to Effort</strong></td>
<td>There as a lack of time between the company and the SPL team. Much rework could be avoid if the stakeholders were together more frequency.</td>
</tr>
<tr>
<td><strong>Requirements and technologies volatility to Effort</strong></td>
<td>Rework was necessary to deal with changes in the features, requirements, and use cases during the project.</td>
</tr>
<tr>
<td><strong>Effort to Motivation</strong></td>
<td>Motivation was an issue for the SPL team because of the effort required to build artifacts for scoping and requirements.</td>
</tr>
<tr>
<td><strong>Communication and Collaboration to Motivation</strong></td>
<td>The team reported that poor communication and collaboration negatively impacted the SPL effort because the SPL team had little feedback at times from the company.</td>
</tr>
<tr>
<td><strong>Iteration and adaptation to Motivation</strong></td>
<td>During focus group, the scoping and requirements team commented about the lack of interaction with other disciplines. They also commented that the long time box of the disciplines (scoping and requirements) reduced their motivation.</td>
</tr>
<tr>
<td><strong>Iteration and adaptation to Standardizing artifacts</strong></td>
<td>When iterations and adaptations were planned, the standardizing specification/documentation for features, requirements, and use cases could be anticipated for the first iterations. For example, an adaptation occurred encompassed some pieces of the textual descriptions inside only one artifact. The decision that originated this adaptation occurred after the first iteration and was adopted in the second one.</td>
</tr>
</tbody>
</table>
Table 4 – continued from previous page

<table>
<thead>
<tr>
<th>Memo Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training to Motivation</td>
<td>The training in the single-systems provided the SPL team with maturity in the use of the features/process. The SPL team suggested that a new role, the product expert, should be created for the scoping and requirements disciplines.</td>
</tr>
<tr>
<td>Company team availability to Communication and collaboration</td>
<td>Communication and collaboration rely on the availability and commitment of the company domain experts. Lack of commitment to the SPL project could cause the project to fail.</td>
</tr>
<tr>
<td>Company documentation to Communication and Collaboration</td>
<td>Reading and mining single-systems documentation were performed in an ad-hoc fashion. The disciplines did not anticipate activities for the analysis of existent documentation. A systematic and efficient process to read and mine existing documentation could reduce the dependency on communication and collaboration with the company team.</td>
</tr>
<tr>
<td>Inspection meetings to Standardizing artifacts</td>
<td>Inspection meetings detected problems in the artifacts and provided new templates to improve the specification tasks.</td>
</tr>
<tr>
<td>Feature, requirements, use cases specification tool to Motivation</td>
<td>The requirements analysts used a tool to specify features, requirements, and use cases (including their variabilities and traceabilities). The stakeholders demonstrated motivation for use of the tool.</td>
</tr>
<tr>
<td>Standardizing artifacts to Communication and collaboration</td>
<td>When the inspection built templates for the stakeholders, the communication through those artifacts improved.</td>
</tr>
<tr>
<td>Iteration and adaptation to Communication and collaboration</td>
<td>Face-to-face communication was more effective for the milestones (presentation of the project results). With more milestones (systematized by iterations with shorter time boxes), increased the collaboration among the stakeholders was identified.</td>
</tr>
<tr>
<td>Company documentation to Communication and collaboration</td>
<td>The company documentation (that was incomplete and inconsistent) aided very little when the SPL team collaborated with company team by documents. Many features were identified wrong.</td>
</tr>
<tr>
<td>New role - product expert to Effort</td>
<td>Interaction with the product expert for single-systems improved understanding of the products by providing information about the systems flow of execution and their features. This streamlined use of the systems, and improved domain knowledge. This fact became more important as the systems was the main information source for features, requirements, and use cases for the SPL team. Effort could be decreased if there was more training.</td>
</tr>
<tr>
<td>Requirements and technology volatility to Iteration and adaptation</td>
<td>When some change took place, the team required short iterations to feedback the adaptations.</td>
</tr>
</tbody>
</table>

Table 4: Reviewed memos.

The following sections answer the case study research questions.

5.2.1. How do the stakeholders characterize the effort to perform SPL SC and RE disciplines?

The stakeholders characterized the SPL SC effort as extensive. This is evident from the amount of documentation generated regarding features,
requirements, and use cases (lines one, two and three, Table 6), divergences on the feature granularity that resulted in a multitude of features (lines four and five, Table 6), and the absence of domain and business experts while legacy documentation was incomplete and inconsistent (line six, Table 6). Moreover, in Table 4, memos relating to effort were observed. Figure 5, shows the cause for the mentioned effort.

The number of features and the absence of domain and business experts were impacted on the effort. To specify features with the appropriate granularity without expertise in the SPL SC discipline and with little knowledge of the domain under investigation was an important factor that impacted on the effort as well.

The effort for the SPL SC discipline can be seen in terms of time for each activity as well. The overall SPL SC discipline took 740 hours and 58 minutes (see Table 5 for more details). The hourly effort is the total
for project participants. It is distributed irregularly as in some cases one participant accomplished the task alone.

In the SPL SC effort distribution (see Table 5), the most costly phase was the product scoping. This took 565 hours and 37 minutes which is 76.33% of the overall discipline execution time. This activity was responsible for the identification, specification, inspection, and validation of the features.

<table>
<thead>
<tr>
<th>Activities</th>
<th>Time</th>
<th>Total</th>
<th>Effort %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Scoping</td>
<td>24h 55min</td>
<td>740h 58min</td>
<td>3.36%</td>
</tr>
<tr>
<td>Domain Scoping</td>
<td>75h 41min</td>
<td>10,21%</td>
<td></td>
</tr>
<tr>
<td>Product Scoping</td>
<td>565h 37min</td>
<td>76,33%</td>
<td></td>
</tr>
<tr>
<td>Assets Scoping</td>
<td>74h 45min</td>
<td>10,09%</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Effort distribution for scoping discipline

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Data</th>
<th>Collection Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC</td>
<td>“... identifying and documenting the features took the major effort...”</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>“...to describe the features was an exhaustive task and took much time to perform...”</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>“...much documentation effort for a small return (value)...”</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>Discordance between the stakeholders about feature granularity</td>
<td>Documentation (SVN)</td>
</tr>
</tbody>
</table>

SC Many features were identified in product scoping (840). The SPL members think that features were described at a low granularity. The product training offered by the company improved understanding about the features.

SC The absence of a domain expert during product scoping, ineffective communication and collaboration with company domain experts, incorrect granularity of several features, and incomplete and inconsistent documentation.

RE “...should prioritize by requirements, features, and/or use cases to optimize the process instead of performing a prioritization by sub-domains...”

RE “...to describe use cases took more time because we had to simulate the operation of the system and describe that observation...”

RE The requirements analyst needs to navigate the system observing the steps to perform that functionality. When the navigation does not work, the requirements analyst needs assistance from company members. This task took a long time (20 minutes).

Table 6: Effort insights.

The stakeholders characterized the SPL RE effort as extensive as well. We deduced this characterization from the high number of features, requirements,
and use cases (line five, Table 6), incomplete and inconsistent documentation from the company (line six, Table 6), and few collaboration between the SPL team and company members (line six, Table 6). Moreover, lines seven, eight, and nine (Table 6) reveal that the effort was impacted by the inappropriate prioritization of requirements, features, and/or use cases; and the description of use cases through the operation in the systems.

As in SPL SC discipline, the number of features and absence of the domain and business experts were factors that the stakeholders should handle during the project. However, to specify step-by-step the use cases (simulation of functionalities, variation points, variants, etc in the systems), incomplete documentation (domain analysts dependency), and prioritization by sub-domain (many features for each iteration) also impacted on the effort.

The requirements analysts carried out the two iterations in 501 hours and 37 minutes (with 144 requirements and 327 use cases specified and inspected). As in scoping, this effort is the total hours for all stakeholders involved in the RE discipline. Tables 7 and 8 show that the use case description is the most costly task, which in both iterations represented 41.5% of the time.

<table>
<thead>
<tr>
<th>Elicit</th>
<th>Specify</th>
<th>Inspect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature</td>
<td>4h 45min</td>
<td>15h 12min</td>
</tr>
<tr>
<td>Requirements</td>
<td>22h 37min</td>
<td>27h 06min</td>
</tr>
<tr>
<td>Use Cases</td>
<td>15h 22min</td>
<td>64h 29min</td>
</tr>
</tbody>
</table>

Table 7: First iteration.

<table>
<thead>
<tr>
<th>Elicit</th>
<th>Specify</th>
<th>Inspect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature</td>
<td>0h 21min</td>
<td>2h 34min</td>
</tr>
<tr>
<td>Requirements</td>
<td>2h 23min</td>
<td>16h 00min</td>
</tr>
<tr>
<td>Use Cases</td>
<td>13h 31min</td>
<td>34h 23min</td>
</tr>
</tbody>
</table>

Table 8: Second iteration.

The inspection process during the RE discipline was another effort observed in the study. The inspection is responsible for the assessment of features, requirements and use cases artifacts for their relationship, their description and their adequacy. This task spent 143 hours and 32 minutes in the first iteration and 139 hours and 23 minutes in the second one.
5.2.2. How do the stakeholders characterize communication and collaboration in the SPL SC and RE?

The stakeholders characterized the communication and partnership between the SPL (SC and RE) team and the company stakeholders as ineffective (see line six, Table 6). This ineffective collaboration stemmed from the observations (in field and interviews) summarized in Table 9 regarding the lack of face-to-face communication during the feature specifications. For example, the company used to validate the artifacts by e-mail (lines seven and eight), since the company employees were busy with other activities (line nine). Moreover, the lack of integration (lines two, four, five, six, eight) was highlighted by the stakeholders. Moreover, in Table 4, memos about communication and collaboration, and Figure 5, show the cause for the mentioned communication and collaboration.

When we look at Table 9, the inception of this characterization is in the line nine, “...SPL team worked on the SPL platform, the company team worked on the WEB ...”. This issue was a factor that the SPL team had to handle during the project. It is evident that the collaboration would be negatively impacted because of this factor.

5.2.3. How do the stakeholders characterize the SPL SC and RE process iterativeness and adaptability?

The SPL SC and RE disciplines were characterized as a waterfall process, due to the very long time boxes (lines three, six, and eight, Table 10). In average, each iteration was two months long (line seven, Table 10).

A group of features for a sub-domain (prioritized by the company domain expert) was used for an iteration. The prioritization resulted in the choice of sub-domains with the most features making the iteration extensive.

The process adaptability was characterized as unplanned. The adaptability emerged only with the developers feelings (in an ad-hoc way), because there was no planning to address it (lines one, two, three, four, and five, Table 10).

In Table 4 and Figure 5, memos show the cause for iterativeness and adaptability.

5.2.4. How do the stakeholders characterize the motivation with SPL SC and RE?

The analyzed documentation shows that the stakeholders (company team) characterized the SPL SC motivation as “satisfied”. However, the SPL team
How the stakeholders characterize the communication/collaboration to perform SPL SC and RE?

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Data</th>
<th>Collection Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC</td>
<td>&quot;... we faced difficulties scheduling meetings with company team ...&quot;</td>
<td>Interview</td>
</tr>
<tr>
<td>SC</td>
<td>&quot;... SPL team was not working together in the same room as the company team. This impacted negatively on communication ...&quot;</td>
<td>Interview</td>
</tr>
<tr>
<td>SC</td>
<td>&quot;... failure to transfer scoping activities to company members ...&quot;</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>&quot;... absence of the scoping expert in the process ...&quot;</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;... there is no integration between SPL and company members ...&quot;</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;... should have someone from the company working together with SPL team ...&quot;</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;... during the specification, communication was performed per chat ...&quot;</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;... moderate involvement, only when we called them to aid us. Sometimes they aided through chat. They did not write features, requirements, or use cases. They participated in some workshops ...&quot;</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>lack of interaction and absence of the customer (company team) in the product scoping phase is the major problem. The SPL team did not integrate well with company members. While the SPL team worked on the SPL platform, the company team worked on maintenance of products and on web technology and features</td>
<td>Observation (in field)</td>
</tr>
</tbody>
</table>

Table 9: Communication and collaboration insights.

characterized the SPL SC discipline as “little motivation”.

In the lines one and two of Table 11, the company team show motivation with the project expressed with the terms “faith” and “satisfied”. On the other hand, the SPL team reveals little motivation (line three) because of other variables (effort, iterativeness, communication, and adaptability).

Characterization of the SPL RE is confused. Some events cause satisfaction with the discipline (lines five and eight, Table 11) and other ones reduce motivation (lines six, seven, and nine, Table 11).

Moreover, in Table 4, memos about motivation, and Figure 5, show the cause for the mentioned motivation.

In general, as an important motivating factor, the SPL team mentioned that trainings on the products, which provided a deeper understanding of the features, increased SPL team motivation (line four, Table 11).

As a demotivating factor, the SPL team reported that the effort to identify the features, their documentation and granularity (which was solved considering the feature concepts in Griss (2000)), no interaction with other disciplines (design, implementation, or testing), ineffective communication
How the stakeholders characterize the SPL SC and RE process iterativeness and adaptability?

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Data</th>
<th>Collection Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC</td>
<td>&quot;...there is no activity for reflections about the process adaptability...&quot;</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>&quot;...the adaptations occurred only in the end of the process...&quot;</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC</td>
<td>Process is named as iterative, but the time box is very long (months), mainly in product scoping phase - The process did not have milestones to reflect about self-adaptations</td>
<td>Observations and Document Analysis</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;...the process does not contain an activity for adaptation...&quot;</td>
<td>Interview (requirements)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;...occurred adaptations (process, format) stemming from the inspection meeting...&quot;</td>
<td>Interview (inspector)</td>
</tr>
<tr>
<td>RE</td>
<td>&quot;...there was iteration but the time box is very long...&quot;</td>
<td>Interview (inspector)</td>
</tr>
<tr>
<td>RE</td>
<td>Two iterations: First iteration from June to October (about five months long). Second iteration from October to December (about three months long).</td>
<td>Document analysis (dotProject)</td>
</tr>
<tr>
<td>RE</td>
<td>very long iterations and absence of the customer (company members) are the main challenges</td>
<td>Observation</td>
</tr>
</tbody>
</table>

Table 10: Iterativeness and adaptability insights.

and the lack of collaboration contributed to demotivation. As the participation of the company team occurred only in validation meeting.

5.2.5. How do the stakeholders characterize SPL SC and RE to deal with volatile requirements and technology?

The SPL stakeholders did not mention in the interviews or focus group that there was a high volatility in the domain, requirements, and technology. However, we noticed volatility in the documentation (bug tracking system) and field observations events for changes such as adoption of a new technology (Web) for development, and adaptive and corrective maintenance in the legacy systems (lines two and three, Table 12).

The SPL stakeholders did not say anything about these events which caused severe changes to the SC and RE disciplines but only reported which features and requirements specifications were updated in the validation and feedback meetings (lines one, three, and four, Table 12).

When the requirements analysts were browsing the system to describe features, requirements, or use cases, doubts emerge that were mitigated with the support of the domain expert. After these events, during the feature, requirements, and use case artifact inspections, the SPL RE steps and artifacts
How the stakeholders characterize the motivation with SPL SC and RE?

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Data</th>
<th>Collection Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC</td>
<td>“...we are motivated with the project so far and we have faith that it will be performed well...”</td>
<td>Interview (company domain expert)</td>
</tr>
<tr>
<td>SC</td>
<td>100 percent of the developers were satisfied using the scoping process. This data was captured in the report of another researcher</td>
<td>Documentation analyzed</td>
</tr>
<tr>
<td>SC</td>
<td>“...little motivation due to repetitive work and we could not visualize the return. Little iterativeness in the process in relation to SPL design and implementation process...”</td>
<td>Interview (SPL team)</td>
</tr>
<tr>
<td>SC</td>
<td>“...the training in the system was productive. Several new features were mapped when the training was held. If we did not have the training, many features would not be mapped...”</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>RE</td>
<td>“…I was motivated with feature specification because I used the feature list from the scoping process…”</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>“…I was not motivated with requirements specification because of misunderstandings between features and requirements…”</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>“…First iteration was less motivating than the second one, because in the second iteration, we had a pattern to specify features, requirements, and use cases…”</td>
<td>Interview (requirements analyst)</td>
</tr>
<tr>
<td>RE</td>
<td>“…we believe in the project, in the activities performed…that the project will be well finished…”</td>
<td>Interview (company domain expert)</td>
</tr>
<tr>
<td>RE</td>
<td>“lack of iterations with others processes (such as design) and unavailability of the customer (company members) in the process caused demotivation”</td>
<td>Observation</td>
</tr>
</tbody>
</table>

Table 11: Motivation insights.

were reviewed in order to make them more effective.

How the stakeholders characterize SPL SC and RE to deal with volatile requirements and technology?

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Data</th>
<th>Collection Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC</td>
<td>“...during the workshops (meetings with the stakeholders) the suggestions (requirements changes) of the company members were updated...”</td>
<td>Interview (developer)</td>
</tr>
<tr>
<td>SC and RE</td>
<td>Many trouble tickets (for adaptive and corrective maintenances) were opened during SPL SC and RE disciplines.</td>
<td>Documentation analyzed</td>
</tr>
<tr>
<td>SC</td>
<td>We noticed that feedback meetings or workshops were used to communicate and update changes in the requirements.</td>
<td>Observation</td>
</tr>
<tr>
<td>RE</td>
<td>The company is concerned with innovations, mainly technology, however, the SPL was not the target of these concerns. According to company’s domain analyst, system functionalities that are not working fostered the changes or updates in the requirements. It was noticed that feedback meetings or workshops were used to communicate and update changes in the requirements.</td>
<td>Observation</td>
</tr>
</tbody>
</table>

Table 12: Requirements and technology volatility insights.
6. Threats to Validity

We considered four criteria for validity, which follows a specific classification scheme (Yin, 2003; Wohlin et al., 2000).

**Construct Validity:** We identified some threats to construct validity and the strategies to mitigated them:

- *Prolonged involvement:* As the SPL project was a new experience for the participants and researchers in real scenario, many events, objects, and their relationships could not be identified and analyzed. To mitigate this threat, we adopted the *prolonged involvement* strategy. The researchers had a close and long involvement with the object of study, which allowed the acquisition of tacit knowledge, enabling avoidance of misunderstandings and misinterpretations (Karlström and Runeson, 2006) on the large amount of information;

- *Data Source Triangulation:* Although there was much captured data on the project, some data source could provide incomplete or inconsistent information. In order to mitigate this threat, we took into account multiple sources, which enhanced the rigor of the research (Yin, 2003) and enabled us to perform a triangulation of data to achieve a greater coverage of research topics (Runeson and Höst, 2009). In this study different sources of evidence were used: interviews, focus group, project documentation, artifacts, and observations to avoid the effects of one interpretation of a single data source; and

- *Peer debriefing:* The main researchers of this study could influence the conclusions because his belief, for example. Thus, we decided to analyze the results in a shared way, where two researchers evaluated the results and another seven participants reviewed them (Karlström and Runeson, 2006).

**Internal Validity.** In our case study, we identified different threats, as follows:

- *Research Questions:* The research questions defined in this study may not focus on the most important aspects regarding agile methods in the context of SPL. We mitigated this risk through discussions with SPL experts, reviewing important papers in the topic, and the mapping study (Silva et al., 2011).
• Interview Questions: The proposed set of interview questions may not have properly covered the Agile SPL area, implying that one cannot find answers to those questions. Maybe, we did not select the optimum set of questions, however we attempted to address the most frequently asked ones and those questions that we considered open issues in the field. We also considered the mapping study (Silva et al., 2011) results regarding agile software product lines to elaborate the questions.

• Observations: The observation was performed by looking at key employees, documentation, and artifacts. However, the most important people and documentation might have not been chosen. To mitigate this threat, the observation was conducted by two researchers.

• Participants Selection: As the selection was based on convenience sampling, the most appropriate set of participants may not have been selected. In this study, the SPL team was selected from different areas with similar background in SPL. The company participants were selected because of their domain expertise.

• Participants Background: The different participant backgrounds may have biased the results because of previous knowledge in SPL. This threat was handled through the selection of a SPL team with little experience in SPL. Two SPL team members had no knowledge in SPL when the project started. The other members only knew SPL concepts through one single academic SPL project. None of the SPL team members had previously applied the adopted processes. When replicating this study, it would be necessary to provide training in SPL for the company team in order to provide a level of experience similar to the our SPL team.

• Company Selection: The selected company may not be the most appropriate, mainly because the company team was not very committed to the SPL project. However, the company had all necessary characteristics to perform a SPL. Its sponsors were interested in reengineering their legacy products into a SPL. Even though the commitment, managerial and technology transfer strategies had impacted on our variables, we understand that this context is very common in industrial scenarios, and therefore, remains interesting to investigate them.
• **External consultants and Hawthorne effect:** Employees often have a problem with external knowledge and consultants that know better. As a result, employees may block this new innovation. Thus, that is a threat that might have biased this study. However, during the case study, there were two ongoing projects being executed in the company. The first was responsible for building the SPL based on the company legacy systems and the other was responsible for legacy system maintenance. The SPL team was only involved in SPL activities and intermittently the domain and business experts interacted with SPL team. Although it is difficult to avoid the Hawthorne effect, the main researcher of the case study and the SPL team was interacting with the company team before this case study began. This strategy meant a more familiar context among the participants was created. Most field observations were performed on the SPL team activities. The adopted data and methodological triangulation could mitigate this threat as well.

**External validity.** As the case study was executed in only one company, it is difficult to make generalizations (Patton, 2001). The findings and discussions in this study are delimited for the SME. Thus, although the findings and discussions could be generalized for large companies, we should not, because the challenges and problems could be different. For example, it is common in small teams for one member to perform more than one role. In our case study, one member had the role of developer, architect, domain expert, and business analyst. However, in large companies, this scenario may not be so common.

Despite the limitations, researchers can extend the study by replicating it in different companies following the context and the design of this study. It enables an analytical generalization, where the results are extended to other cases, which have common characteristics and hence for which the findings are relevant, i.e. defining a theory.

**Reliability.** It is concerned with to what extent the data and the analysis are dependent on specific researchers. Hypothetically, if another researcher later on conducted the same study, the result should be the same (Runeson and Höst, 2009). The use of a guideline (Runeson and Höst, 2009) for both design and report the case study, as well as, the definition of a case study protocol (Brereton et al., 2008) and a structured case study database with all the relevant data can mitigate this threat. In our study, the case
study protocol, interview protocol, excerpts of the focus group meeting, and some notes about the field observations were defined and they are available in a public web site\(^4\). The database with all relevant data is not available publicly because of company confidentiality rules.

7. Discussion

This section discusses the identified weaknesses when characterizing the SPL SC and RE disciplines. To better organize and facilitate the understanding of this section, we grouped the weaknesses regarding to the following items: (i) inadequate technology transfer strategy, (ii) lack of company commitment with the SPL project, and (iii) inappropriate managerial strategy factors. These are detailed in five subsections (c.f. Section 5.2). In addition, this section presents the identified challenges and future directions when performing the SPL SC and RE disciplines, the lessons learned, and the main finding of the project.

Before we delve into the upcoming subsections, it is important to mention that we did not state any expectations in terms of effort, communication and collaboration, iteration, adaptation, motivation, and volatility prior to this study. This case study is the first industrial experience for the SPL team, thus our intention is to characterize the SPL SC and RE disciplines while understanding their weaknesses to justify the use of agile methods with SPL SC and RE.

7.1. Effort weaknesses

The technology transfer strategy, company commitment, and managerial factors impacted on the effort. Several weaknesses were identified and we grouped them according to these factors.

7.1.1. Technology transfer strategy

The technology transfer strategy was a factor that fostered the others to a certain extent. In the adopted strategy, there was little integration between the SPL and company teams, because the SPL team performed the SPL development, while the company team only validated its artifacts. Company team was focussed on other project that was already in progress.

\(^4\)http://riselabs.dcc.ufba.br/ivonei/case_study/
7.1.2. Company commitment

Low integration between SPL and company teams fostered a weak commitment to the SPL project.

The lack of company commitment to the SPL project resulted in the absence of domain and product experts during the SPL SC and RE disciplines, which could impact on the features, requirements, and use case specifications.

To mitigate the absence of domain and product experts, the SPL team elicited and inspected the features, requirements, and use cases through legacy systems. However, an important problem with the legacy systems during the elicitation was the dependence on the participant understanding and abstracting the product functionalities. The validation meetings identified understanding and abstraction mistakes in the features, requirements, and use cases artifacts. The fine granularity issue mentioned in Section 5.2 is a result of these mistakes. It may have contributed to the high number of features.

We believe that better interaction between the company domain and product experts and the SPL team could improve the use case specification resulting in reduced effort. This is reinforced by the opposite effect that communication and collaboration had on the effort in the causal loop diagram, Figure 5. In addition to the memo Company team availability to Communication and Collaboration (Table 4) which addresses the company commitment with the SPL project.

It is also possible that the immaturity of the company team in the software engineering practice areas (Jones and Northrop, 2010) affected the company commitment factor. This may be due to the SPL SC and RE disciplines requiring the use of the formal software engineering practices such as requirements elicitation, documentation, and inspection which were not desired by the company team.

7.1.3. Managerial

The managerial factor of the adopted approach was not appropriate. There was little feedback, mostly received by email and with a long time box between validations.

The managerial factor also fostered the effort weaknesses for the prioritization rule used in the iterations that did not work properly.

The prioritization rule consisted of prioritizing the sub-domains to be first used in the SPL SC and RE disciplines. As each sub-domain was composed of many features with its associated requirements and use cases specifications,
each iteration had many artifacts that had to be built, making the effort to perform any iteration exhaustive. In addition, as the iterations were adjusted to meet the sub-domains, the time boxes became very long. The sub-domain grouping and prioritization caused exhaustive iterations for the SPL team. With the grouping and prioritization by features, requirements, or use cases with shorter time boxes, it could increase the iterativeness and foster process adaptability earlier. Thus, the effort would be decreased for each iteration.

In the causal loop diagram, Figure 5, it is possible to see that the iteration and adaptation variable has an impact on the effort. Thus, the time box reduction would address the SPL team point of view. This would be in accordance to the memo Iteration and adaptation to Motivation (see Table 4) and the eighth insight in Table 6.

7.1.4. Technology transfer, commitment, and managerial

We observed that features granularity and difficulty in differentiating features and requirements are associated with each other. These aspects impacted on the effort as well.

As can be seen in the memo New role - product expert to Effort, in Table 4, and in the causal loop diagram, Figure 5, we can decrease the effort in SPL SC and RE, if we had performed the training in the legacy products. Training could improve understanding of the features and requirements.

This training provided a deep understanding regarding the company domain. As the product expert (employee who had strong experience in training users and customers in the products) showed practice situations performed by users and customers to face daily activities. The training clarified several questions for the SPL team related to the product operation and domain definitions. As a result, the number of features was reduced from 3644 to 840 features.

7.2. Communication and collaboration weaknesses

The weaknesses in terms of lack of inadequate technology transfer strategy, company commitment with the SPL project, and inappropriate managerial strategy for the project also contributed to ineffective communication and collaboration between company and SPL teams.

7.2.1. Technology transfer strategy

To mitigate the weaknesses on communication and collaboration, an extractive model (Krueger, 2002) with with an incremental or pilot technology
transfer strategy, which reuses one or more existing software products for the product line could be more appropriately to foster interest from the company.

It can be seen from Figure 1, that in the company product portfolio there is a product that encompasses the other products. Thus, this model seems to be more appropriate for our context and the idea of an incremental strategy could work well.

This model as an incremental or pilot strategy could contribute to some actions. For example, both teams could focus on the same features, technologies, and disciplines, especially, if the results were executable artifacts (code of functionalities, visual prototypes) scheduled within the SPL project iterations. Thus, the SPL team could be closer to customers and increase the face-to-face interaction with the company domain experts.

7.2.2. Company commitment

We could observe the negative impact of the company commitment on the communication and collaboration in the following evidence: (i) in Table 9, the memo Company team availability to Communication and collaboration in Table 4, and the causal loop diagram, Figure 5, where the company team availability has constraints on the communication and collaboration; (ii) in Table 9 the company focused on maintenance and migration of single-systems/Web, paying little attention to the transition to SPL; (iii) in Table 9 the participant of the case study declared the lack of integration between the SPL and company teams.

7.2.3. Managerial

The weaknesses regarding the company commitment and managerial factors resulted in two main weaknesses for communication and collaboration. First, the size of the iteration time box was long, thus there was a reduced number of communication and collaboration events during the SPL SC and RE disciplines. Second, as there was few interactions among the participants, the feedbacks basically occurred during the validation meeting, when the iteration had been completed.

During an iteration, when some questions emerged, the communication and collaboration among the stakeholders were seldom performed face-to-face to solve the issue.

7.3. Iterativeness and adaptability weaknesses

The inadequate technology transfer, lack of company commitment with the SPL project, and inappropriate managerial aspects of the project factors
identified weaknesses in the SPL SC and RE disciplines in terms of iterative-
ness and adaptability (see Table 10).

7.3.1. Technology transfer strategy

The inadequate technology transfer strategy (proactive and top-down
with poor integration between the teams) impacted on the project two fold.
Firstly, on the company commitment which was a pre-requisite for adapt-
ability and iterativeness. Secondly, inadequate management as in long time
boxes and poor self-reflections regarding activities, roles, and artifacts.

As a suggestion, to mitigate these issues in the project, an extractive
technology transfer strategy is more appropriate with agile approaches that
can improve the adaptability and iterativeness in the processes, strong de-
developer commitment and appropriate management to enable the adapta-
tions and iterations.

7.3.2. Company commitment and managerial

In the scoping discipline, the main identified weakness was that only one
iteration occurred and no other discipline was performed. In requirements
engineering, two iterations occurred, however, many features, requirements,
and use cases specifications were performed per iteration. During the focus
groups with the SPL team, they suggested weekly iterations with different
disciplines, such as design and implementation.

The SC and RE disciplines did not provide any suggestions for adjust-
or improvement of the processes (see Table 10). This is the main weakness
in any approach. It cannot adapt itself to the company context, team, and
unpredictable changes, in order to become more effective. In other words,
the team had to cope with other weaknesses without changing them.

Both iterativeness and adaptability have direct impact on effort, moti-
vation, and communication and collaboration variables in the causal loop
diagram (see Figure 5). Thus, the teams should address feedback about it-
eration and adaptation in the next steps of the SPL project (Tables 4 and
10).

7.4. Motivation weaknesses

All the weaknesses previously mentioned impacted on the motivation dur-
ing process execution. In this section, the most important weaknesses (in
terms of the previous variables, effort, communication and collaboration, and
iterativeness and adaptability) are reported.
In both SPL SC and RE disciplines there was a large effort to perform some tasks. We noticed excessive descriptions on some features, for example, descriptions very similar for the glossary, features, requirements, and use cases specifications. During the SPL SC and RE inspection activity, the inspector described some problems with non-conformities such as incompleteness and ambiguity on the features, requirements, and use cases. The superficial understanding of the domain and legacy systems, lack of a specific template, and misconceptions regarding the specification task were the root cause for the non-conformities. Maybe, glossary, features, and requirements descriptions could be just one artifact. This would greatly reduce the effort in the SPL RE discipline.

Even though the company was located close to the team, its employees were busy with the single-systems evolution (legacy system) and paid little attention to the SPL project impacting on the communication and collaboration.

The long iterations (see Table 4) prevented the stakeholders visualizing the artifacts achievements in future activities such as design, implementation, and tests. We inferred that the participants of this study believed that short iterations could provide insights and improvements for SPL engineering knowledge.

Fourth, there was no adaptability in the disciplines. Even though the domain is relatively stable, it is necessary that the disciplines can be adjustable.

A positive impact on motivation was generated by the use of the SPL tool (Cavalcanti et al., 2011), which helped the stakeholders during features, requirements, and use cases elicitation and also to maintain the traceability among these artifacts. However, the tool still requires improvements to be as effective as a CASE tool that integrates features modeling to architecture and testing activities.

By observing the field, we noticed an alignment between the company activities, such as source code evolution, and activities for the SPL such as executable prototypes, including the possibility of reuse of source code. These could positively impact on motivation.
7.5. Volatile requirements and technology weaknesses

The company has been developing products in the information systems for medical management domain for more than 15 years. Thus, their products had maturity regarding implemented functionalities. However, as any domain, volatility can be present for features or technology.

As the SPL team did not have contact with the company customers, any major volatility in customer features could have been hidden from the SPL team. However, investigating the bug tracking system, there were many change requests for adaptations and corrections. These requests are indicative of some kind of volatility. An additional set of studies have been undertaken towards understanding how these legacy systems evolve, and how the volatility might impact the SPL project.

7.6. Challenges and future directions

Through analysis, we identified the main challenges regarding the weaknesses mentioned. These challenges suggest new directions for dealing with SPL SC and RE disciplines. We verify if the process effort, communication and collaboration improvements, process adaptability, company commitment, managerial, SPL scalability, and technology transfer strategy challenges can be addressed by the new approaches for dealing with SC, RE, and other disciplines. Our comments are as following:

- **Process Effort:** The effort for requirements engineering could be reduced by the use of agile practices such as test-driven development (TDD) (Williams et al., 2003). Using TDD, the acceptance and unit tests, and user stories (Beck, 1999), could replace or support the activities which involve features, requirements and use case elicitation, specification, and validation (Ghanam and Maurer, 2008, 2009; Ghanam et al., 2010; Ghanam and Maurer, 2010a,b). As the aforementioned practices are agile, the process could be iterative and incremental and, for example, the SPL architecture could emerge from the tests. Refactoring and incremental design could also suitable in this context.

- **Communication and Collaboration Improvements:** The use of the single-system documentation could have compensated for the lack of communication and collaboration among the stakeholders (John, 2010). The existing documentation should be mined in order to answer the simple questions about the legacy systems. As a result, work load of the domain experts could be reduced as well.
The use of collaboration engineering patterns (Briggs and Grünbacher, 2002; Briggs et al., 2003) in the SPL context can be an opportunity for the stakeholder collaboration process because they can support the stakeholders involvement during the scoping discipline.

Basically, the use of those patterns would address significant agile principles for the scoping and requirements disciplines such as (i) “Business people and developers must work together daily throughout the project” and (iii) “At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly” (Noor et al., 2008, 2006).

Another opportunity to deal with this challenge is the agile practice also applied in studies with SPL, planning game (Carbon et al., 2006, 2008). It addresses tasks prioritization in agile software development. In the SPL project, it can be used to prioritize features, requirements, or use cases and support the interaction between core assets and application developers, as the practice encourages communication and collaboration among the stakeholders.

- **Process Adaptability:** In the case study context, the SPL team identified two aspects to be adjusted, the iteration time box and the relationship of the SC and RE with other disciplines. To perform these two adaptations some aspects should be considered such as organizational culture and technology. In addition, other aspects could be identified, what lead us to question how we could set adaptability on the RE and SC disciplines.

The SPL and company teams should embrace the project mission. The approaches should be flexible enough to adapt to different contexts and projects characteristics. An important action, that provides information regarding process adaptability, would be the feedback through reflections on the process, fast and continuous meetings with retrospective thoughts on the iteration, and improvements the stakeholders learned incorporated into the process.

Some practices applied in Feng (2008) could also be replicated in other projects with requirements elicitation such as on-site interactive sessions from the extreme programming (XP) method and white boards from the LEAN method.
Methods such as dynamic systems development method (DSDM) (Stapleton, 2003) and feature driven development (FDD) (Palmer and Felsing, 2001) which embrace adaptability in a natural way could be used Tourret (2006); Paige et al. (2006). The DSDM defines the business study such as where the scoping and the high level reference architecture could be implemented. The FDD defines an activity to build an overall model for a specific domain, where the scoping, requirements and reference architecture could be addressed.

SCRUM could be also be useful for assisting the adaptability issues. The daily meeting and sprint retrospective could be applied to improve learning and discipline adaptation (Raatikainen et al., 2008).

- **Company Commitment, Managerial, and Technology Transfer Strategy:**
The adopted technology transfer strategy was completely against the values, principles, or practices widespread in agile or lean approaches. For example, all the Lean principles such as *Go See, Help partners be lean, and Pull* (Larman and Vodde, 2008) and agile principles such as *Customer Collaboration and Business people and developers must work together daily throughout the project* should be tried until the teams identify the best way to build the SPL project.

The agile or lean principles also impact the company commitment and managerial factors. The principle *Go See* - weekly, for instance - could require greater company commitment to the SPL project. Regarding management, the SPL team could change the iteration prioritization rule e.g. by features. The management could try an iteration pilot with a small time box, less requirements and use cases, and involvement with other disciplines such as architecture, implementation, and testing. The feeling that the SPL project brings value to the company business could increase.

- **SPL Scalability:** Due to number of features (840) and the fact that requirements plus use cases were specified based on the previous set of features, we had a high number of requirements plus use cases as well as many interactions with the domain analysts to elicit and validate these artifacts.

To minimize this problem, the SPL team performed some prioritizations considering the sub-domains. The SPL team also used a tool
(Cavalcanti et al., 2011) to manage all the artifacts, the variability, and the traceability among them. However, prioritizations on features and a comprehensive tool could foster the scalability in a more appropriate way.

7.7. Lessons Learned

By conducting the case study and analyzing the results, we identified some lessons learnt. They are described next.

The inspection activity evaluated the quality of the artifacts built. It also aided in mitigation of problems that emerged during the tasks. The activity occurred in two iterations. The first iteration had a huge impact in the second one, as the lessons learnt from the first were considered during the specification of the new artifacts. Thus, future iterations will be performed with more quality and less effort according to the memo Iteration and adaptation to Effort recorded in the Table 4.

Besides the previously described activities, an important task was the use case specification with variability. As advocated by Gomaa (2005), the SPL team adopted the same use case elements to describe the scenarios and variabilities, such as alternative flow and extended use case. The only difference is the use of tags to identify where in the flow (main or alternative) a variation point can occur and a tag for the element that enables traceability with requirements artifacts. There are different practices to describe use cases in SPL contexts (Eriksson et al., 2005; Moon et al., 2005). However, further research is required to decide which ones are more efficient.

Other lessons learnt are the need for a system expert role to support the SPL team in the understanding of products functionalities; the training intertwined with the iterations; the use of guidelines to specify features, requirements, and use cases suggested by inspection; and the support tool to manage the assets such as features, requirements, and use cases.

In addition, we understand that the limited availability of domain experts in general is inevitable and it is intensified in SME context, where the domain experts can assume different roles. This unavailability fosters an important question to be explored further “How can a SPL project decrease the dependency on domain experts?” This question can guide us to conduct scoping and requirements in legacy systems using different sources, such as documentation and code.

Finally, requirement and use case artifacts could be replaced by others from agile methods. Stories and backlog items could replace requirements as
they can describe the requirements and the variabilities. In the same way, the acceptance test might replace the use case, as it can describe scenarios for requirements.

7.8. Main Finding

In this case study we highlight some general findings.

First, although the company is considered small, it had communication problems among stakeholders. These problems and those mentioned about iterativeness, adaptability and volatility negatively impacted on effort and motivation.

Second, the adopted scenario and the identified weaknesses can support the integration of agile methods into SPL SC and RE disciplines, as the agile practices and principles address those bottlenecks and there is evidence to support combining both approaches (Silva et al., 2011; Diaz et al., 2011).

Third, the motivation can be improved through the mitigation of the variables effort, communication and collaboration, iterativeness, and adaptability with an extractive, incremental, and pilot strategies.

Fourth, the extractive model with a strategy for SPL incremental or pilot could be more appropriated to foster interest from the company.

8. Conclusions and Future Work

SPL engineering provides a systematic way to reuse common artifacts to meet customer needs, by developing different products. SPL Scoping and Requirements Engineering are important disciplines, which capture the business essence and domains of the company. During these two disciplines, several bottlenecks emerged in the project.

We grouped them according to three weaknesses factors: inadequate transfer strategy, lack of company commitment, and inappropriate managerial strategy.

The case study discussed these weakness factors under five different variables: (i) effort in applying SPL disciplines, (ii) project communication and collaboration, (iii) iterativeness and adaptability, (iv) motivation, and (v) requirements and technology volatility.

The identified bottlenecks that impacted on effort are: the absence of domain and product experts causing large dependence on participant understanding and abstraction on product functionalities; the prioritization rule
used in iteration prioritizing in the sub-domains; the difficulty in differentiating features and requirements; and feature granularity. Besides, communication and collaboration, iterativeness and adaptability variables also affected the effort.

In terms of communication and collaboration, the bottlenecks are: the size of the iteration time box was long; there were few interactions among the participants, thus the feedbacks mostly occurred during the validation meeting; and the lack of face-to-face communication and collaboration among the stakeholders.

In terms of iterativeness and adaptability, the bottlenecks were: only one iteration occurred and no other discipline was performed; many features, requirements, and use cases specifications per iteration were performed; the disciplines did not provided any aspect for adjusting or improvement of the processes; and the poor self-reflections regarding to activities, roles, and artifacts.

All the previous bottlenecks impacted on motivation. The bottlenecks on the volatility variable are the same as observed in the iterativeness and adaptability variables. In this case, whether there is no iterativeness and adaptability, it is difficult to deal with changes because those provide milestones for adjustments and identification of problems with the process, for instance.

These bottlenecks were not handled by the SPL SC and RE disciplines. This fosters challenges for future work.

The study results discussed some emerged challenges. Challenges that may be overcome with some changes such as more participation of the domain experts, shorter iterations in the SC and RE disciplines, and iterations with other disciplines. However, the improvements can be better carried out with approaches that have presented successful cases in this direction such as the Agile SPL approaches.

The use of agile approaches in the SPL context has demonstrated important academic and industrial results in the area (Silva et al., 2011), even though focusing on working code, and reducing up-front design and the process overhead of traditional approaches (Cockburn, 2001).

In future work, we intend to conduct other case studies to collect further empirical evidence and apply agile practices and principles into the SPL framework.
References


of the international conference on Agile methods in software development, pages 27–34, Orlando, USA.


