PhD Theses in Experimental Software Engineering

Vol. 65

Philipp Johannes Diebold

Agile Practice Experience Repository for Process Improvement

FRAUNHOFER VERLAG

PhD Theses in Experimental Software Engineering
Volume 65

Philipp Johannes Diebold

Agile Practice Experience Repository for Process Improvement

Fraunhofer Verlag

Zugl.: Kaiserslautern, TU, Diss., 2019

Printing:
Mediendienstleistungen des
Fraunhofer-Informationszentrum Raum und Bau IRB, Stuttgart

Printed on acid-free and chlorine-free bleached paper.

ISBN (Print): 978-3-8396-1555-3
Fraunhofer-Informationszentrum Raum und Bau IRB
Postfach 800469, 70504 Stuttgart
Nobelstraße 12, 70569 Stuttgart
Telefon +49 711 970-2500
Telefax +49 711 970-2508
E-Mail verlag@fraunhofer.de
URL http://verlag.fraunhofer.de

Agile Practice Experience Repository for Process Improvement

Vom Fachbereich Informatik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation
von

M.Sc. Philipp Johannes Diebold

Fraunhofer-Institut für Experimentelles Software Engineering IESE
(Fraunhofer IESE)
Technische Universität Kaiserslautern

Berichterstatter:	Prof. Dr. Dr. h. c. Dieter Rombach Dr. Richard Turner (SEI)
Dekan:	Prof. Dr. Stefan Deßloch
Datum der wissenschaftlichen Aussprache:	12. Juli 2019

D 386

Acknowledgments

In the first place, I would like to thank my principal advisor Prof. Dr. Dr. h.c. Dieter Rombach from the University of Kaiserslautern and the Fraunhofer Institute for Experimental Software Engineering IESE. His help in the form of continuous advice and feedback greatly improved my thesis. Furthermore, I am very grateful to Dr. Richard Turner as my second advisor, who supported me mainly in the final year with valuable feedback and discussions as well as by suggesting concrete improvements for my thesis.

Furthermore, I gratefully acknowledge the funding received towards my PhD from the German Federal Ministry of Education and Research (BMBF) in the context of the funding program "Software Campus". Within this project, Bosch Chassis System Control (CC) was the non-funded application partner from industry that provided insight into their practices, gave me the opportunity to try different things, provided feedback, and published papers in collaboration with me. Therefore, many thanks to Dr. Peter Fröhlich and Dr. Udo Mayer from Bosch.

Additionally, my PhD work was supported to a large extent by "ProKoB", a research project on the specification, description, and impact of process building-blocks. For this reason, my gratitude goes to all the project partners who are working with us in that project, namely Frank Simon, Cordula Schmidt, Yves Rausch, Johannes Wahl, Christoph Kemmann, Oliver Kemmann, and Andreas Scharf.

Next, I would also like to thank several people at Fraunhofer IESE for supporting me and bearing with me when I was stressed from working on my PhD over the last few years. From the scientific point of view, the help of Dr. Jens Heidrich, Dr. Andreas Jedlitschka, and Dr. Liliana Guzman was particularly invaluable: They introduced me to people, arranged meetings, and provided valuable feedback on my thesis and the related work. From the practical point of view, Thomas Zehler, Sven Theobald, and Lena Sembach provided input, cooperatively worked with me on the approach, and were available for discussions at any time. Furthermore, I thank Constanza Lampasona and Anna Schmitt for their discussions and guidance during my PhD. I would also like to acknowledgment Sofia Vidal, Marc Dahlem, and Dominik Richter, who worked for me as student research assistants during my PhD.

In addition to being grateful for the support I received at Fraunhofer IESE, I would also like to thank Dr. habil. Daniel Mendez Fernandez, Stefan Wagner, Darja Smite, and Marco Kuhrmann, who supported me during my PhD work. They especially helped conduct the ImpAct workshops, which are strongly related to my PhD approach.

Fraunhofer IESE and my Software Campus project gave me the opportunity to go to New Zealand on a research exchange, which enabled me to get involved in cooperation with colleagues from there. Thus, I would specifically like to thank to Austen Rainer, Stephen G. MacDonnell, Jim Buchan, Matthias Galster, and Sherlock Lickorish for their discussions regarding my PhD and for their cooperation in some initiatives that resulted in joint publications.

Finally, I especially want to thank my family and friends for supporting me during my PhD. My gratitude goes to my friends Malte Brunnlieb, Peter Reuter, Thilo Rauch, Jan-Gerrit Schettler-Köhler, Markus Kleine, Jürgen Spengler, Michael Rampetsreiter, Ricarda Daub, and Alexander Jürgens. Last but not least, many thanks to my parents Anne and Martin, my brother Jakob, for their patience and continuous support during the long road to my PhD. This project would not have been completed without your support.

To all the people who (continuously) asked about the current status of my PhD: Now the never-to-be-changed answer is that it is done! "It's done when it's done." (Agile Guru)

Tena rawa atu koe!

Abstract

Context: Agile development and specifically Scrum are prevalent in the software industry, especially in information systems units. Agile development is well accepted, frequently adapted, and recognized for increasing change manageability, transparency, communication, productivity, time to market, and team morale. These demands are also increasingly common in embedded domains. As a result, interest in agile development is also rising in these domains, but their need to comply with regulations presents an obstacle to the adoption of agile development.

Problem: For practitioners, the appropriate degree of agility, defined as the set of agile methods or practices adapted, is mostly identified ad-hoc or based on gut feeling. The current processes, independent of whether they are agile or not, do not sufficiently address the latest improvement goals. Furthermore, the existing literature shows that there is a lack of evolutionary agile transition approaches. Finally, knowledge is missing about the application of agile elements to (regulatory) constraints.

Objective: To solve these problems, a decision support approach for evolutionary agile transition shall be developed by systematically collecting, documenting, and using information on agile practices.

Contributions: This thesis introduces the **A**pproach for goal-oriented and **C**ontext-specific **A**gile **P**rocess **I**mprovement (ACAPI) using agile practices to support the agile transition or transformation. It consists of the *Agile Potential Analysis* and *Simulation of Process Improvements*. The *Agile Potential Analysis* is built on the Agile Practices Impact Model, a conceptual model for the impact of agile practices. The aim of the analysis is to fill the transition backlog with identified suitable practices based on the improvement goals and the context, e.g., regulatory constraints. A *Simulation for Process Improvements* using a Monte-Carlo simulation extends the analysis to obtain more quantified data.

Goal achievement, comprehension, and acceptance of ACAPI were evaluated in several industrial case studies from different domains. The validation demonstrated that all three aspects were improved using the approach, with the highest effect on the comprehension.

Table of Contents

List of Figures

List of Tables

1 Introduction

"People don't adopt a methodology
they adapt it"
Tom DeMarco

Starting with the motivation for the topic of this thesis (Section 1.1), this chapter presents the research approach (Section 1.2), which addresses the problem statement (Section 1.3). The research approach is detailed by describing the research contributions with the aligned hypotheses (Section 1.4). Furthermore, some background on agile development is given (Section 1.5) and the overall structure of the thesis is described (Section 1.6).

1.1 Motivation

Several decades ago, software development started becoming ever more comprehensive and complex, evolving into an engineering discipline called Software Engineering (SWE) (Sommerville, 2012). Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software (IEEE Computer Society, 1990). Since this discipline covers many different aspects, it is divided into fifteen sub-disciplines. One of these sub-disciplines is that of software development processes, which encompasses the definition, implementation, assessment, measurement, management, change, and improvement of software (life-cycle) processes (IEEE Computer Society, 2014). As in every engineering discipline, the continuous improvement of development processes, called Software Process Improvement (SPI) (Münch, et al., 2012), is very important as process environments continue to evolve, e.g., through new laws and regulations or customer standards, which may influence or impact the processes themselves.

One of the core ideas of SPI is continuous improvement (Münch, et al., 2012), which is often performed as a time-boxed initiative when problems appear, such as insufficient quality or too much time between releases. If such time boxes are applied continuously, the result is evolutionary improvement. Compared to revolutionary improvement (Diebold & Zehler, 2016), it is smoother and considers the current process. The examples of problems mentioned above show why SPI is most often performed in a goal-oriented manner. The most common improvement goals in this context include product quality, time to market, employee satisfaction, and others (Diebold & Schmitt, 2016).

The field of research regarding development processes started with the creation and evolution of different lifecycle model, such as the Waterfall lifecycle model (Royce, 1970) or the Spiral model (Boehm, 1988), and was refined by concrete processes such as the Rational Unified Process (RUP) (Kruchten, 2003). These models describe the traditional way of developing or engineering software and are therefore called plan-based processes. They are more commonly known as rich or structured processes due to certain properties such as fairly detailed planning, largely complete and detailed requirements specification, and sophisticated architecture (Diebold & Zehler, 2016). Snowden and Booned (2007) showed with their Cynefin Framework that these processes are needed in certain contexts.

Since the turn of the millennium, another field of processes has appeared besides rich processes: agile processes or approaches. A lot of new methods evolved in this field (Abrahamsson, et al., 2003). In 2001, the core values of all these new methods – focus on individuals, working software, customer collaboration, and responding to change – were enshrined in the Agile Manifesto (Beck, et al., 2001). Since that time, the use of agile approaches in software development processes as well as agile project management methods started to grow fast, and today agile development is very common (VersionOne, 2018) (Komus & Kuberg, 2016) (Kuhrmann, et al., 2018). Scrum is the dominating method (VersionOne, 2018), but there are a lot of other elements in agile development (Abrahamsson, et al., 2010).

The majority of companies using agile development are working in domains related to information systems, such as web development, mobile development, telecommunications, media & entertainment, and finance (VersionOne, 2018). Even though popular studies do not (explicitly) consider regulated embedded domains, experience shows that the propagation of agile development is significantly lower in regulated domains (Kugler Maag CIE, 2015) (Diebold & Theobald, 2018).

Definition: Regulated Domains

A regulated domain is a specific field that is influenced and restricted by external regulations. Common regulations are laws, standards (e.g., ISO, DIN, etc.), or assessment models from customers or other stakeholders. Examples of regulated domains are Automotive, Avionics, Space, Railway, Medical Devices, Defense, and Banking.

However, companies from regulated embedded domains experience similar problems, such as the need for shorter time to market, higher product quality, or higher customer involvement (Diebold & Theobald, 2018), which are issues that can be resolved or at least addressed by agile development (VersionOne, 2018). In order to also benefit from the advantages of agile development, companies from embedded domains have recently (in the past ten years) also started using agile development to some extent, using, e.g., certain elements of agile development or using agile approaches only in early phases of their development.

This transition towards more agility is still ongoing. Its completion is impeded by the large number of organizational constraints found in these domains, from external and internal regulations, such as standards or laws, to complex structures, e.g., caused by the large size of an organization or by supplier-contractor relationships. These aspects make it harder to use agile development, respectively to adopt it in the first place (Diebold, et al., 2015).

1.2 Background on Agile Development

When software development evolved into an engineering discipline, development processes become more and more important (Sommerville, 2012). These systematic approaches, which define detailed activities covering the whole lifecycle, are called rich processes because they focus on repeatability, predictability, extensive documentation, up-front system definition, and detailed plans (Boehm & Turner, 2003) (Diebold & Zehler, 2016).

Besides this family of processes, around the turn of the millennium agile software development appeared and became a counterpart to these traditionally used rich processes. With the manifesto of agile software development (Beck, et al., 2001), its foundations were documented with their four core values as well as the underlying principles. These core values are: (1) individuals and the interactions between them, (2) working software, (3) strong customer collaboration, and (4) response to change. These core values and the underlying principles both focus on the mindset and culture of how to do development.

Even before the Agile Manifesto, initial versions of Scrum (Schwaber & Sutherland, 2017) and XP (Beck & Andres, 2007) had appeared as new development methods. As the most common methods in this field, they were further developed and improved and many other approaches or methods followed (Abrahamsson, et al., 2010). Thus, the new class of agile methods emerged.

Definition: Agile Methods

Agile methods (= agile processes, agile approaches) are methods that define how software or systems are developed over the whole lifecycle or major parts of it using a specifically named set of agile practices. (Diebold & Zehler, 2016)

Similar to software engineering methods in general (independent of their application field), agile methods are defined in specific details, building especially upon a set of very detailed and fine-grained practices as defined above. The most common examples of such agile practices are the twelve core practices of XP, which initially define a set of practices as the XP method. Most of the other agile methods do not specify the practices in detail but define them implicitly (e.g., Scrum) through different roles, artifacts, and events.

Definition: Agile Practices

Agile practices (= agile techniques) are established instructions, e.g., tasks, activities, technical aspects, or guidelines, with a specific focus or with an aspect in the development of software that is performed according to one or more agile core values and agile principles. (Diebold & Zehler, 2016)

As already indicted in the previous paragraphs, the two cultural aspects of agile development (core values and principles) are linked with the technical aspects (methods and practices) in the following way (also depicted in Figure 1): The foundation for the technical agile aspects used, such as agile methods and agile practices, is the Agile Manifesto. Here, the core values form the basis for the more detailed and refining principles. Whereas agile methods are aligned with the idea of agile development because they mostly cover the whole lifecycle and several or all principles, agile practices are generally used only for certain parts of the lifecycle; therefore, they might not always be aligned with all the principles and core values.

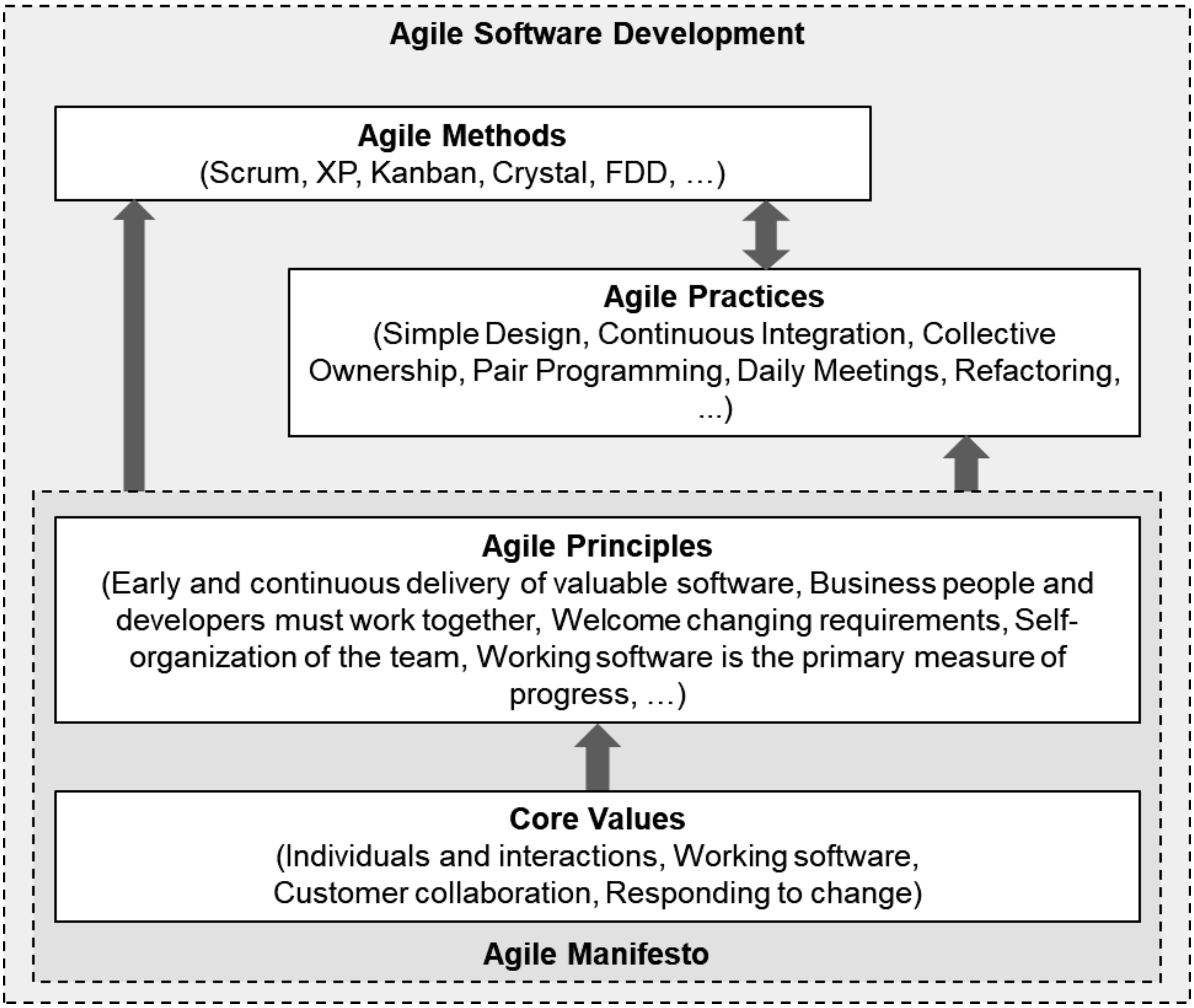


Figure 1: Agile World - Connection between cultural (core values and principles)) and technical agility (practices & methods) (Diebold, et al., 2015)

Cultural agility – referring to core values and practices – and technical agility – referring to methods and practices – are coupled and influence each other (Diebold, et al., 2015). Every technical agile element has some cultural prerequisite that it needs to unfold all of its potential. Nevertheless, most of the elements can also be integrated with a less agile mindset; e.g., it would be possible to initially establish Daily Stand-Ups as a continuous calendar event. Furthermore, every implemented agile practice contributes to the agile culture and thus often positively influences the mindset. In the example of the Daily Stand-Ups, after the event including its three questions has been used for some time, it should be established well enough that it will no longer be necessary to include it in the calendar. This shows the interaction between technical and cultural agility.

1.3 Problem Statement

Based on the empirical study presented in Chapter 2 regarding the current and future usage as well as the introduction of agile development, the following two practical problems (PP) and two scientific problems (SP) were identified:

PP1 – The current process does not sufficiently address improvement goals: When we investigated currently existing problems in our state-of-the-practice survey (Diebold & Theobald, 2018), we found several organizational and process problems that could be addressed by agile benefits or that would provide reasons for introducing agile development.

PP2 – Ad-hoc selection of appropriate agile elements: In software process improvement, especially when it is about improving processes by using agility, the improvement actions, i.e., the agile practices or methods, are often selected ad-hoc based on gut feeling. At the beginning of an agile transformation or introduction, mostly only popular methods such as Scrum are known and consultants are paid to perform the selection based on their experience.

SP1 – Lack of an evolutionary agile transition approach: Even if most consultants promote a revolutionary / big-bang approach of introducing agile methods, our study showed that stepwise integration of agile development is favored as a way of introduction. However, except for the very generic and high-level solution of performing goal-oriented software process improvement, there is a lack of support for an evolutionary agile process improvement or transition approach. This is the case because most agile consultants are like evangelists promoting complete agile methods to be implemented in a big bang, although often a hybrid process would actually combine the best of both worlds.

Definition: Evolutionary Agile Transition

In terms of (software) process improvement, evolutionary agile transition is a step-by-step change process consisting of small individual steps. It uses the current As-Is situation as a starting point and moves towards an (intermediate) goal that is related to technical agility (e.g., methods and practices) and/or cultural agility (e.g., principles, values, and mindset).

SP2 – Lack of knowledge of how agile elements affect (regulatory) constraints: In software process improvement, considering the organizational and project-specific context is essential. Thus, it is important to know what agile elements are applicable in which contexts as well as to understand their behavior, benefits, and drawbacks in this specific context. It is also necessary to know which improvement actions contribute to what kind of contextual aspects. Particularly in regulated domains, these aspects are becoming more and more important because organizations are often forced to comply with different external regulations and their regulatory constraints. Without knowing how agile elements will impact their specific constraints, companies will either not embark on such an improvement journey or may go in the wrong direction. In both research and practice, such knowledge has started to grow over the last two to three years, but it is not sufficient yet, especially in domains such as Automotive.

1.4 Research Scope

During an agile transition or transformation (cf. Figure 2), the following levels are passed:

- Agile Team (single team)
- Agile Product or Project (project or product development with several teams)
- Agile Organization (complete organizational unit with all projects or products (e.g., individual business units))
- Agile Enterprise (complete enterprise with all units)

The work done within the scope of this thesis took place at the team and project levels, but some of the improvement suggestions made might also influence the organizational levels.

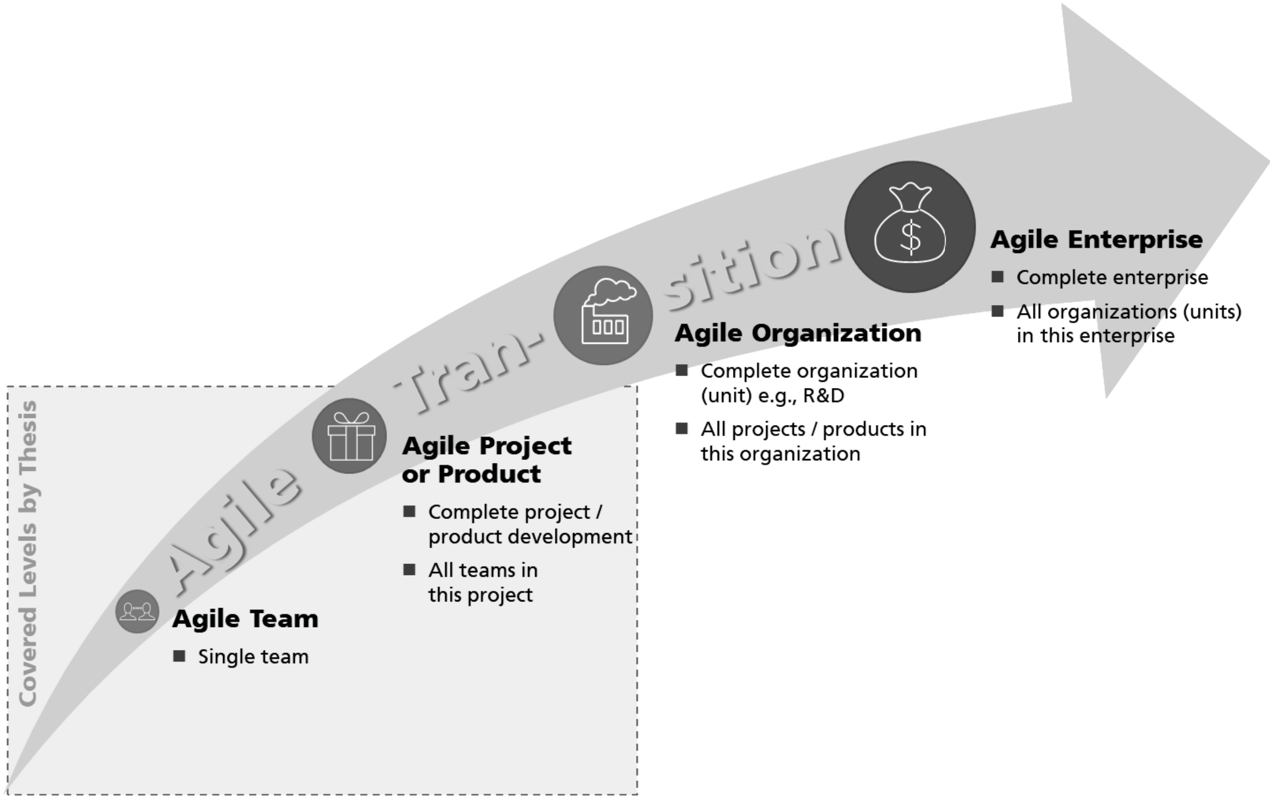


Figure 2: Agile transformation

Furthermore, it is important to mention that the implementation of the agile transition is beyond the scope of this thesis. The approach presented here provides the input actions to start the transition. It does not provide support during the transition, unless new input should be given along the way.

Even though the scope of this thesis is not limited to any specific domain, it does have a strong focus on regulated embedded systems and software domains. This is due to the fact that information systems do not have that many problems with the application of pure agile methods and are already quite saturated in this regard. Within the mentioned domain scope, the examples in the thesis will especially focus on the Automotive domain because this is one of the main application domains of Fraunhofer IESE.

1.5 Research Goals and Contribution

1.5.1 Goals

Motivated by the four problem statements listed above, the following research goals of this PhD thesis were derived. Figure 3 depicts the connection between the problems and the research goals.

RG1 – Higher goal achievement with new or improved process: Based on the fact that the current process often does not address problems or improvement goals (PP1), we aim to create an approach that proposes concrete practices to be integrated into the process in order to realize higher goal achievement.

RG2 – Increasing decision-making confidence: Another goal is to increase decision-making confidence throughout the whole evolutionary agile transition (SP1). This goal is mainly, but not exclusively, derived from the problem of ad-hoc selection of agile elements (PP2).

RG3 – Making the impact of agile elements on (regulatory) constraints explicit: Especially in the face of external control regarding specific mandatory regulations, the need exists to make the impact of agile elements, i.e., agile practices, on specific (regulatory) constraints explicit (SP2). Besides the aspect of concrete and given regulatory constraints, the goal also includes the consideration of common contextual constraints and how they affect agile elements.

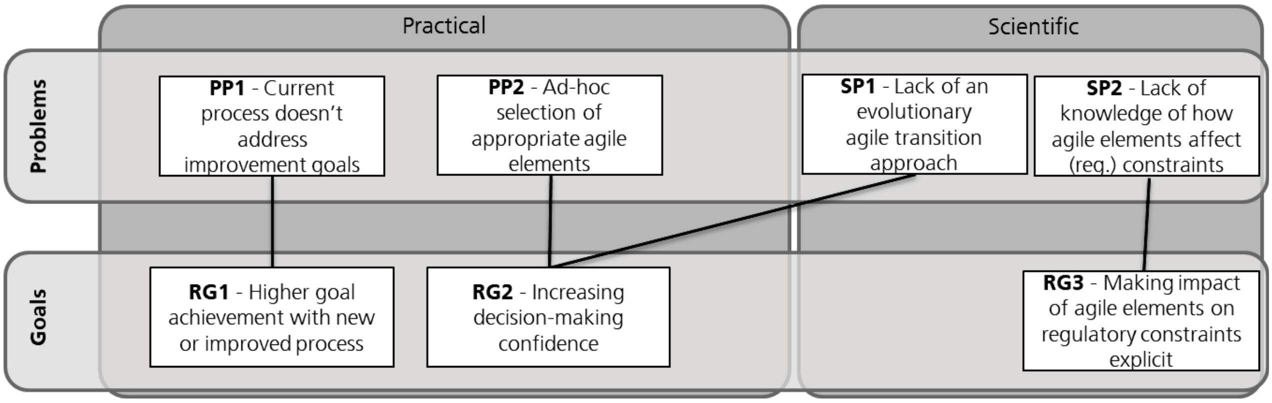


Figure 3: Mapping of problems to research goals

1.5.2 Research Methodology & Solution Idea

The research goals imply the need to create a goal- and context-specific method (RG1 & RG3) using some kind of experience repository (RG2) as a foundation.

The research methodology includes four phases as shown in Figure 4: (1) state of the practice, (2) state of the art, (3) development of the solution approach, and (4) evaluation of the approach. The following paragraphs will summarize the scope and goal of each step. Detailed descriptions can be found in the corresponding chapters.

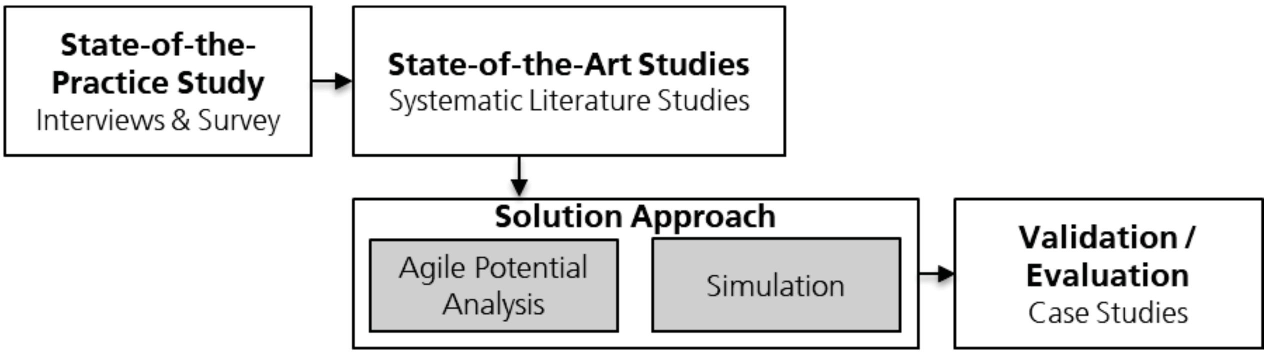


Figure 4: Overview of research methodology

State-of-the-Practice Study: To characterize the state of the practice of agile development and process improvement specifically in regulated domains, we designed and conducted an empirical interview and survey study (Chapter 2). Besides the current state of agile usage, state-of-the-practice problems and requirements regarding agile process improvement were identified.

State-of-the-Art Studies: To understand the state of the art in software process improvement, especially in agile process improvement, we performed two studies together with colleagues:

(1) a systematic mapping study on software process improvement in general with a broad scope;

(2) a detailed systematic literature review covering only the aspects identified as agile software process improvement.

Development of Solution Approach: The solution approach is a team- or project-specific analysis method used as a decision-support mechanism for filling an agile transition backlog. This overall approach is subdivided into two methods best performed in combination: *Agile Potential Analysis* and *Simulation of Process Improvements*. Both parts can be combined and work at a fine-grained level, suggesting the use of single agile practices. The development of the approach started with the identification of possible and necessary input. Using that input and knowledge about existing approaches in other areas (not necessarily software engineering), the two methods were created and iteratively refined.

Evaluation / Validation: Both parts of the overall approach were included in the evaluation performed as part of this thesis. Six case studies on the *Agile Potential Analysis* were conducted with respect to the first two research goals (RG1 and RG2). For the *Simulation of Process Improvements*, one case study was performed with the focus on decision-making confidence (RG2).

1.5.3 Research Contribution

Based on the identified problems from practice and research as well as the research goals, the contributions in this PhD thesis can be distinguished into foundation & model building contributions (C1 – 2), methods (C3 – 4), and tooling aspects (C5):

C1 – Agile Practice Repository. As a foundation, a repository of agile practices was built. It contains a large number of practices, including all the practices from Scrum and eXtreme Programming, the most frequently used agile methods (VersionOne, 2018). All of them are described according to a given schema. However, the most important part of this repository is the Agile Practice Impact Model, a causal model for modeling the impact of agile practices. This model aims at visualizing the positive as well as the negative effects of agile practices on different process characteristics, such as transparency or communication.

C2 – Simulation Model. Based on the Agile Practice Impact Model and the collected quantitative data, a simulation model was created. In particular, the simulation model integrates the quantitative data on the impact of Agile Practices in order to enable detailed quantitative analyses.

C3 – Agile Potential Analysis. This analysis method was created to provide decision support regarding the identification or selection of appropriate agile practices for process improvement. This step-wise analysis uses improvement goals as well as context information as input and applies the Agile Practice Repository (C1) to select appropriate improvement actions, i.e., agile practices.

C4 – Simulation Method. Around the simulation model (C2), a method was designed that uses the model and runs it with different configurations such that the simulation results can be used for a detailed analysis. It was developed especially as an addition to the Agile Potential Analysis to provide more insights into possible context changes.

C5 – Tool Support for Potential Analysis. Additionally, some tool-related contributions were also developed: To support the Agile Potential Analysis (C3) with its underlying amount of data, two different kinds of tool support were developed. An Enterprise Architect (EA) plugin containing all the needed elements is available as modeling support for the analysis. This enables detailed analyses due to its ability of easily reduce complexity. Furthermore, an Excel Macro was developed based on all the impact data of the repository, which can easily provide intermediate outputs of the analysis.

C6 – Empirical Contributions. Furthermore, this thesis makes some empirical contributions to agile software engineering research and practice because several studies were conducted in its context: On the one hand, the Agile Practice Repository, and particularly its causal model, was filled with the results of a longitudinal study using interactive posters for data collection as well as some case studies. On the other hand, an evaluation was performed of the approach, respectively of its two parts. Six industrial case studies were conducted for the *Agile Potential Analysis* (C3), while the *Simulation of Process Improvements* (C4) was validated by one case study.

1.6 Thesis Structure

This thesis contains six chapters and two appendices. It is structured as follows:

Chapter 2 describes the state of the practice regarding the different aspects related to this thesis: It includes the usage of agile development in general, in regulated domains, and in the Automotive domain as examples. Furthermore, knowing how agile development is introduced in companies is important for our work. Based on these two aspects, we summarize the practical problems found and derive the requirements necessary to address them.

Chapter 3 first presents the state of the art of Software Process Improvement (SPI). A more detailed look is then taken at the agile SPI approach in order to identify all existing approaches to compare them with the approach developed in this thesis. The description of the major approaches is followed by a discussion and comparison with regard to the practitioners' requirements (cf. Chapter 2).

Chapter 4 details the initial solution idea in the development of our methodological approach. The two parts of the integrated method, the *Agile Potential Analysis* and the *Simulation of Process Improvements*, are described, as well as their integration and interaction. The description includes the models they use, the process steps, examples, tool support, and many other aspects.

Chapter 5 presents the empirical validation of the newly developed agile process improvement approach. Besides the hypotheses and the plan for an ideal validation, this chapter includes the empirical studies performed for validating parts of the approach. For the *Agile Potential Analysis*, six different industrial case studies were performed. In contrast, the *Simulation of Process Improvements* part could only be validated with one case study using a walkthrough.

Chapter 6 summarizes the results and contributions. Furthermore, it discusses limitations, open questions that arose from the thesis, as well as potential future work regarding the different parts of this thesis: methodology, empirical validation, and tool support.

2 State of the Practice

"The world is full of willing people, some willing to work, the rest willing to let them."
Robert Frost

This chapter presents information about the state of the practice in agile development (Section 2.1) and the introduction of agile development (Section 2.2). Furthermore, the overall practical problems stated in the Motivation (cf. Section 1.1) are derived from this state and refined into concrete requirements (Section 2.3).

2.1 Using Agile Development

The current state of agile development in this section is divided into different aspects. We will start with a general overview and three studies; two common surveys on agility and one systematic mapping study (Section 2.1.1). Next, in Section 2.1.2, we will discuss the topic of agile development with respect to its application in regulatory domains based on an empirical study conducted by us. In Section 2.1.3, we will finally focus specifically on the state of agility in the Automotive domain, a regulated domain that is our example application domain for the approach developed in this thesis.

2.1.1 Agility in General

The most prominent study on agile development is VersionOne's "Annual State of Agile Survey" (VersionOne, 2018), which was conducted in the 11th edition in 2016. With about 4,000 responses, it is one of the largest surveys covering all different sizes of organizations from all over the world (dominated by North American countries). The list of domains to which their participants belong shows a strong focus on information systems, with only Healthcare (6%) and Transportation (2%) as embedded systems. Despite this contextual data, the report does not provide any detailed results for specific groups.

The report states that 95% of its respondents practice agile development. However, when we take a closer look at this aspect, we find that only about 10% of them claimed that all of their teams are working in an agile manner. In terms of the time they had already been working using agile methods, 40% were rather new, whereas the others had already been using agile development for a longer time.

The main reasons for adopting agile development were: "accelerate product delivery", "enhance ability to manage changing priorities", "increase productivity", and "enhance software quality". The top benefits and improvements from implementing agile development were "manage changing priorities", "increase team productivity", and "increase project visibility".

Regarding agile methods, nearly 70% of the survey respondents practice Scrum or Scrum/XP hybrids (58% plain Scrum, 10% hybrids), which also had an impact on the most frequently used agile practices, such as daily stand-up, backlog, short iterations, retrospectives, as well as iteration and release planning.

Another survey dealing with the "Status Quo of Agile" is (Komus & Kuberg, 2016). The second run in 2014 included 612 participants from more than 37 countries (with a focus on Germany with more than 60%) and 20 different domains. Considering the different domains in detail, all the embedded domains (in their case Automotive, Healthcare, Automation, and Aerospace) sum up to less than 10%, similar to the previous study.

Overall, their results showed that 15% of the participants only used traditional processes. This means that 85% of the participants were working in an agile manner, of which the minority mentioned that they were completely agile (21%). Most worked in a hybrid way (39%) or selected one of the two ways depending on the specific context (25%). Two thirds of the participants had been working in an (at least partially) agile way for the last four years (the majority, 21%, had started in 2012).

The most frequently used agile method was Scrum, followed by Kanban, Extreme Programming (XP), and Feature Driven Development (FDD). Similar to the results of the previous survey, the usage of Scrum influences the most commonly used agile practices: daily scrum (89%), user stories (81%), product backlog (80%), sprint review (79%), and sprint backlog (78%). The only practice not originating from Scrum but often used in addition to Scrum is user stories.

Regarding improvements achieved with agile development, the creators of the survey provided the following six categories: deliverable quality, employee motivation, teamwork, adherence to schedule, efficiency, customer orientation, transparency. Regarding these categories, they rated the most common agile methods and compared them with traditional processes and project management. All agile (and lean) methods performed better. The participants rated Scrum best, especially concerning product quality, motivation, teamwork, and transparency.

In addition to these two survey studies, we performed a systematic literature study on the usage of agile practices in practice (Diebold & Dahlem, 2014). Even though we tried to identify the domain of the publications in order to elaborate more on domain-specific aspects, almost a quarter of the publications found did not specify any information on that. In those that did specify the domain, information systems domains such as Finance & Insurance or Telecommunications were the dominant ones.

Due to the fact that this study focused only on single small agile practices, which are also covered by the two previous surveys by at least one question, a comparison and aggregation is possible and helpful. The results of this literature study showed time boxing, planning meeting, learning loop, specification practices, product vision, and daily discussions (such as daily stand-ups) as the most frequently used practices. The results also show that there were three practices in particular – quality checks, pair programming, and customer involvement – that were used either partially or had been adapted to some extent.

Although this literature study considered abstract agile practices, its results can be compared with the results of the previous studies. This comparison, which was partly done in (Diebold, 2014), showed that all the different studies present similar results: The agile practices used most frequently are the Scrum practices, but the numbers also show high usage of variations of Scrum, as shown in (Diebold, et al., 2015) because not all Scrum practices are used equally often. Furthermore, the results show common usage of add-on practices to agile methods, such as user stories or test-driven development.

2.1.2 Agility in Regulated Domains

This section presents parts of a survey conducted by Diebold and Theobald to evaluate the state of the practice of development processes with respect to the usage of agile development in the context of regulated embedded domains from the perspective of practitioners (2018). For the data collection, we used a combination of semi-structured interviews, including some open questions to facilitate open conversations, as well as a subsequent questionnaire based on the interview guidelines. During the interviews, both interviewers took notes independently, which were later compared and aggregated in a spreadsheet and integrated with all the results. Based on the experience from the interviews, the guidelines used for creating the questionnaire were slightly extended with a question using a

Likert-like scale because there were too many answers to the open questions.

Regarding the response rate, which we can only report for the interviews, 24 of the initially contacted 49 contacts ultimately participated in the study. Since we had a total of 50 participants in this study, the remaining 26 came from the follow-up online questionnaire. The interview participants all worked in German-speaking countries, mainly in Germany. As intended, our participants worked in very different embedded domains:

Domain	Interviews	Survey	Total	Percentage
Automotive	9	8	17	28.3%
Avionics	2	6	8	13.3%
Medical	3	13	16	26.7%
Space	2	1	3	5%
Defense	3	3	6	10%
Railway	3	4	7	11.7%
Governance	-	1	1	1.7%
Machine Building	2	-	2	3.3%

Table 1: Demographics of the study on agility in regulated domains (Diebold & Theobald, 2018)

Before we discuss agility in regulated domains in detail, it is interesting to observe that according to the respondents, the lifecycle process used is most often a combination dominated by the traditional V-model (n=34) or waterfall model(n=18). These two are combined with iterative development (n=21) and/or agile development (n=24). The V-model is dominant as it has the character of a framework for documentation and is often demanded by the customer, e.g., in the aerospace domain. The number of combinations of traditional aspects with iterations and/or agile aspects was quite high and matches the results regarding hybrid processes (Kuhrmann, et al., 2015) (Kuhrmann, et al., 2018).

A detailed look at agile usage reveals that most participants mentioned restricted usage (n=32) due to specific context factors, such as specific projects with less hardware, early phases of a project such as research or pre-development, or less critical projects. For the same reason, most companies (n=28) selected the approach that fits best. Agile was mainly selected where the customer demanded/accepted it (n=6) or where requirements were volatile (n=5). Eleven participants stated that they always use their given agile approach. One common example given by one participant is the usage of Kanban in maintenance projects.

For those who used agile development at least to some extent, the experience in terms of number of years using it varies a lot: Most were quite new, with three years or less (n=20). Some had used it for between three and five years (n=8) or between five and ten years (n=9), while only four had used it longer than that. Participants reporting more than ten years of usage either had a long transition or claimed to do agile development mainly based on the iterative nature of their processes.

Similar to the results of Section 2.1.1, Scrum was the most frequently used method (n=21), followed by adapted Scrum (n=9) and Kanban (n=7). Some participants did not explicitly mention a specific agile method and reported that they only used single agile practices (n=6). The practices and concepts most often mentioned were being iterative (n=7); having short iterations, incremental development, or pair programming; holding daily meetings; and using backlogs (all n=4). Fast feedback, retrospectives, and planning meetings (all n=3) were also used by the participants. The most common adaptation of Scrum was the definition of add-ons (n=10), such as additional roles, practices, or improved documentation. Examples of roles are "architect", "Scrum team lead", or "project manager to protect team". Reasons given for the increase in documentation included "Requirements, architecture and design specification has to exist based on regulations" or "Risk management and traceability of requirements must be documented". Furthermore, modifications were made regarding different aspects. The duration was changed, for example, "dailies take more time", "extended to 30 min, if necessary, sometimes longer", "sprint length varies" (n=5) "between 1 and 6 weeks", or the frequency was changed. Regarding roles and responsibilities, the participants involved in meetings changed (n=5), such as specific "architecture stand-ups", or "product owner breaks down tasks" (n=5). Further details on adaptations can be found in (Diebold & Theobald, 2018).

Besides being interested in current usage, we were also interested in how the participants see the future with regard to agility. For the future aspect, we were mainly interested in the answers of those participants who were employed in companies not yet working in a completely agile way: 19 of the participants currently partially using agile development and five not using it at all see its usage as their future. Ten would like to keep their status of only using agile development to some extent due to their existing and future context, which does not allow using it completely. Furthermore, two people not using agile development at the moment wanted to use it partially in the future. With the exception of only one person who wanted to move from partially to none, all the participants would like to keep their current status or increase their degree of agility (which was mentioned by the majority).

2.1.3 Agility in Automotive

As the main application domain used within this thesis will be the Automotive domain, we will briefly discuss the usage of agile development of our automotive data and the "Agile in Automotive – State of Practice" study (Kugler Maag CIE, 2015).

Compared to the overall dataset of (VersionOne, 2018), agility seems to be more commonly used in the Automotive domain than in regulated domains in general, since there were more participants from this domain in the study who claimed to be working in an agile manner compared to those using agile development only partially. Nevertheless, partial usage of agile development is still predominant. Even if these aspects differ slightly vary from the overall data, the agile elements used (methods and practices) as well as their common adaptations are more or less the same. Only because of specific regulations such as Automotive SPICE (VDA QMC

Working Group 13, 2015) or ISO26262 (International Organization for Standardization, 2011) were the major add-ons in this domain dedicated roles required by such regulations. Similar to the overall usage, the future usage was also rated slightly better in the Automotive domain compared to the overall data. For example, all of the Automotive participants envisioned a future with agility. Although complete agile development was mentioned more often than partial use of agile methods and practices, the participants from the Automotive domain were less confident about completely using agile development everywhere.

The "Agile in Automotive" study showed that agile development is used more frequently in pure software-related processes than at the system level. In addition to that, it showed that the highest usage is in the Multimedia domain, which is less safety-critical than the Automotive domain with regard to most aspects. Another aspect of agile usage throughout the automotive development cycle showed that the focus is on the initial phases of the development[1]. When we compare these results with the results of our study, they mutually confirm each other, even though (Kugler Maag CIE, 2015) presented some more Automotive-specific details.

Similar to its dominance in overall agile usage as well as in regulated domains, adapted Scrum dominates agility in the Automotive domain. The study mentions that it is rarely used by the book, but rather in combination with other methods such as XP. Furthermore, our study agrees with (Komus & Kuberg, 2016) in that Kanban is often used in the maintenance and support of systems and software development projects. Considering detailed agile practices, the major focus on Scrum also means that Scrum practices were dominating in their study, including practices such as daily stand-up, backlog, short iterations, retrospectives, and iteration planning. Furthermore, (Kugler Maag CIE, 2015) also provide more information on the adaptations, which were also additions in their case. The two most

[1] In the Automotive domain, these are also called samples (A, B, C, D), which provide components as prototypes with increasing functionality and integrate them into test vehicles.

frequently added roles are quality product owner and safety product owner. Most often, the Automotive domain tries to keep the traditional roles and extends them with the responsibilities of agile roles. This is also the reason why the Scrum roles heavily diverge from the original ones of the Scrum guide.

2.2 Introducing Agile Development

Besides the already mentioned and discussed usage of agile development or agility, in the context of this thesis it is important to know more about the introduction of agility, such as why and how agility is introduced.

In our study, the major reasons for introducing agile development were related to time (n=26), volatility (n=12), process quality, customer involvement/collaboration, and internal collaboration (each n=5). These reasons for introduction can also be compared with current and existing problems. In most cases, these expectations were achieved, even though some of the participants mentioned that success was only partial. For those who had not started their agile journey yet, the expected benefits show a similar distribution, from time (n=22) via process quality (n=9) and volatility (n=7) to communication (n=6).

Since the focus of this thesis is on regulated environments, we were interested in how much regulations influence the introduction and usage of agile development. None of the participants considered agile not usable due to the regulations. Since only three believed that regulations do not have any influence on agile development, the majority believes that regulations do have either a weak influence (demanding extensions or adaptations to agile methods, n=11) or a strong influence (only allowing the introduction of single agile practices, n=9) on agile development. This also reflects some of the interview statements, for example that agile development "has to be implemented within a traditional corset". When we only consider the data collected from the Automotive domain, the weak influence was even more dominant than in the overall set.

The final and most important question was how agility should be introduced in a dedicated context, such as a team, project, or a complete company. The stepwise (sometimes also evolutionary) integration of agile development was clearly favored over the big-bang approach. The majority (n=20) clearly considered the stepwise approach to be better, and some tended towards this approach (n=12). The minority of the participants tended towards the big-bang approach (n=5) or noted it as their favorite (n=3). This distribution was more or less the same when we looked only at the Automotive domain. In addition to this quantitative data, some participants made statements such as: "Change can only happen slowly"; “a stepwise approach helps to build acceptance", "changing the complete established process causes fear of change" and "is related to a higher risk". Furthermore, one particular participant reported process problems after their big-bang change, especially with regulations.

2.3 Problems and Requirements

In the following, the requirements derived from the problems identified in our study will be listed and explained, so that they can be used for discussing and comparing the related approaches.

2.3.1 Current Problems

In the study presented above (Diebold & Theobald, 2018), we started each interview by asking the participants about their current development problems, independent of their current degree of agility. Based on these problems, we were able to identify the most prominent problems as well as to check how agility might contribute to solving them.

The problems described in (Diebold & Theobald, 2018) were coded, resulting in the following distribution, which is presented in Figure 5 (left red columns). The problems mentioned most frequently – independent of whether in the interviews or in the survey – were: process quality (n[2]=30), development time (n=22), product quality (n=13), standard conformance (n=10), communication (n=9), customer/user involvement (n=8), documentation (n=8), and volatility (n=7). Considering only the interviews, where the majority of the participants came from the Automotive domain, additional problems related to internal collaboration (n=13), (process) integration (n=11), and roles/skills/responsibilities (n=13) were stated quite often.

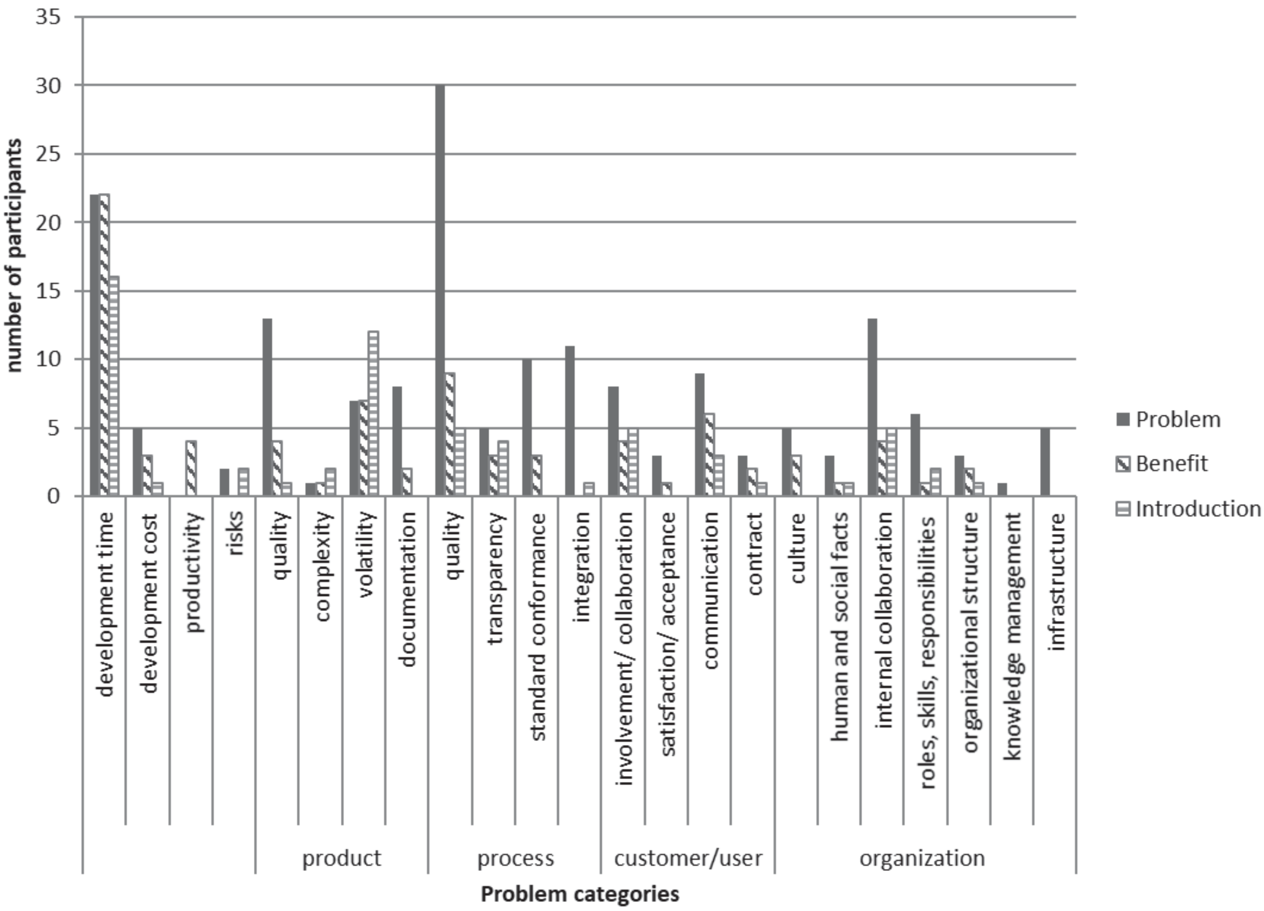


Figure 5: Current development problems (left columns) compared with agile benefits (center columns) and the introduction of agile development (right columns) (Diebold & Theobald, 2018)

[2] representing the number of mentions

The aspects illustrated in Figure 5 and discussed in the previous paragraphs confirm that the current process does not address improvement goals (PP1, see Section 1.2) due to the wide range and large number of existing problems mentioned. In our specific case of agility, it is even more interesting to see how the experiences and expectations with agile development match the above-mentioned problems. For this reason, Figure 5 (center blue and right green columns) presents a comparison of these two aspects. Development time as one of the major problems is also the top issue that is addressed or covered by agile development. Process quality is by far the most frequently mentioned problem, but only some process quality aspects seem to be covered by agile development. This is the case even though process quality is the third most frequently covered problem in agile development, since only development time and product volatility are covered to a greater extent.

2.3.2 Requirements

In order to enable better discussion and traceability of all the requirements described below, we categorize them as follows:

We start with **goal-/problem-related requirements**, which cover the requirements regarding the issue of considering current problems or improvement goals. These are followed by **evidence-based requirements**, which focus on the basis on which the decisions in the approaches are made. **Continuous SPI requirements** cover the focus of the respective approach, reversibility, and the details for providing input on improvements. Finally, in addition to all these topic-related requirements, there are **context requirements**, as we need a context-specific approach that takes into account the specific regulated context that is our focus.

After the presentation and detailed explanation of the individual requirements, the requirements will be traced and related to the problems described in Section 1.3 and the research goals stated in Section 1.5.1.

Goal-/Problem-related SPI Requirements

REQ1: Relationship to (organizational) goals. In measurement, it is a common approach to align measures with organizational or project-specific goals or derive them from these, e.g., by using GQM (van Solingen, et al., 2008) or GQM+Strategies (Basili, et al., 2014). Since the idea of process improvement is the same, we came up with the requirement that the improvement suggestions are related to the (organizational) goals (Diebold, et al., 2017).

REQ2: Stakeholder involvement. Similar to the overall agile idea of having strong customer involvement as well as cross-functional teams including designers, user experience specialists, requirements engineers, developers, and testers, involving as many stakeholder as possible and necessary is very important. Especially in combination with the existing problems and current improvement goals, it is important to get the views of the different stakeholders.

Evidence-based SPI Requirements

REQ3: Transparent decision-making. As transparency is one of the major parts of the agile culture, this requirement demands that any improvement decision needs to be transparent to all the stakeholders interested in the decision and should provide the rationale for why and how it is to be made.

REQ4: Evidence repository. In conjunction with transparent decision-making, the existence and usage of an evidence repository for agile process improvement is mandatory. Without such a repository, decisions will be made based on gut feeling or on one's own experiences, and experiences cannot be shared across persons, teams, projects, or event organizations.

REQ5: Use of quantitative data. Most existing case studies mainly cover complete agile methods such as Scrum or XP. Existing evidence on single practices, however, is often of a qualitative nature because it is the experience of experts. Many people would prefer basing their decisions at least to some extent on quantitative data about the impact of a particular practice.

Continuous SPI Requirements

REQ6: Step-by-step evolution. "Change is only possible slowly" or "change of complete process is utopia" (Diebold & Theobald, 2018) were given as statements when we asked the participants of our study about the integration of agile development. Therefore, we consider step-by-step evolution as a requirement for SPI and especially for agile development.

REQ7: Reversibility of changes. Along with step-by-step evolution, it is also important that the process improvements that are performed are easily reversible. This is necessary, for example, in cases where an improvement did not work or has a negative influence and the alleged "improvement needs" to be rolled back.

REQ8: Suggesting improvement actions. This requirement considers the aspect that the identified problems or improvement goals should not only be discussed, but that actual improvement actions should be proposed.

Contextual Requirements

C-REQ1: Consideration of regulatory requirements. Especially in embedded domains, companies are required to comply with different types of regulations. Most of them are related to processes, with some being actual process-related standards such as Automotive SPICE (VDA QMC Working Group 13, 2015), while others are indirectly linked to processes, such as safety standards. These standards are only one example of regulatory requirements. All such regulatory requirements need to be considered in an approach.

C-REQ2: Consideration of context issues. Independent of the concrete regulatory requirements, software engineering and especially the field of development processes have shown to be highly depend on context. Moreover, since agile development or a specific agile method or practice is no silver bullet, a solution method needs to consider the concrete context, from team to organizational level (which might include the above-mentioned regulations) (Diebold, et al., 2017).

2.3.3 Mapping Requirements to Problems and Goals

As already specified in the descriptions of the different requirements, all of them were derived from the initial practical or scientific problems. Since the overall goals of this thesis are derived from these problem, the requirements can also be related to them. These relations are presented in Table 2 below:

Table 2: Mapping agile SPI requirements to problems and goals

Requirements	**PP1**	**PP2**	**SP1**	**SP2**	**G1**	**G2**	**G3**
Goal-/problem-related SPI Req.							
Relationship to organizational goals	X				X		
Stakeholder involvement	X	X			X	X	
Evidence-based SPI Req.							
Transparent decision-making		X				X	
Evidence repository		X		X		X	X
Use of quantitative data		X				X	
Continuous SPI Req.							
Step-by-step evolution			X			X	
Reversibility of changes			X			X	
Suggesting improvement actions		X	X			X	
Context Requirements							
Consideration of regulatory req.				X			X
Consideration of context issues		X		X		X	X

Table 2 shows that every requirement was derived from at least one problem and therefore belongs to at least one goal. We derived most requirements from the problem of ad-hoc selection of appropriate agile elements (PP2), followed by the two scientific problems: lack of an evolutionary agile transition approach (SP1) and lack of knowledge about the impact of agile elements (SP2). The two goal-related requirements belong to the problem that the current process does not address improvement goals (PP1). The relationships between the problems and goals were used to derive the requirements mapping to the three improvement goals.

2.4 Summary

Summarizing this chapter, we started with the state of the practice with regard to the usage of agile development on different levels:

- Agility in general
- Agility in regulated domains
- Agility in Automotive

Especially the studies in regulated domains in general and in the Automotive domain in particular show the potential and the need for the use of more agility to provide benefits such as management of changing priorities, project visibility, or delivery speed.

Next, we described how agility is introduced in specific contexts. Even though the revolutionary big-bang introduction is predominant in practice, most people favor the evolutionary way, which is similar to common change initiatives. Finally, we discussed current problems in practice, such as process quality, development time, or product quality, and derived different requirements for the approach of this thesis:

- goal-related requirements
- evidence-based requirements
- continuous SPI requirements
- contextual requirements

(VersionOne, 2018) and (Komus & Kuberg, 2016) show a saturation of agile development in the software industry. But our study as well as others focusing more on the embedded domain, respectively on the specific domain of Automotive, showed that there exists a need in these domains for the use of agile development, as it is not widely used yet in these domains and could serve to address several of the problems encountered in these domains.

3 State of the Art

"All men by nature desire knowledge."
Aristotle

This chapter includes information and an overview of the state of the art related to this thesis, which focuses on Software Process Improvement (SPI) in general and on agile SPI in particular. To this end, a systematic mapping study was performed covering any kind of SPI approaches (Section 3.1), which was further refined by a systematic literature study on the specific topic of agile SPI (Section 3.2). The detailed and related approaches will be discussed in detail and related to the problems and requirements.

3.1 Systematic Mapping Study on SPI Approaches

In this section, we present a short summary of a systematic mapping study (SMS) on the state of the art of software process improvement (Kuhrmann, et al., 2016). Our research approach is based on (KItchenham & Charter, 2007) for the systematic mapping study and on (Petersen, et al., 2008) for the systematic literature review, which is explained in detail for our case in (Kuhrmann, et al., 2016).

For this overview study on the domain of SPI, our idea was to get a picture of existing topics and to investigate research trends over time. Therefore, we defined the following **questions** for this study:

- *What is the general publication population on SPI?* This question aims at getting an overview of the general publication pool on

SPI, e.g., publication count, frequency, and different research type facets addressed by the publications found.

- *What is the contribution population?* Based on the publications found, this question is aimed at the topics addressed and the major contributions (e.g., SPI models, theories, secondary studies, and lessons learned).

- *What trends can be observed in SPI and SPI-related research?* This question aims at investigating the focal points addressed by SPI research to date and at identifying gaps as well as trends for the future.

These three questions were asked in this overall study, but this thesis will only focus on the third research question regarding the trends, especially the trend towards agile SPI. The results for the other questions can be found in (Kuhrmann, et al., 2016), which also reports on the detailed results of the **data collection procedure**, such as query construction, data sources, and data format.

Besides the commonly collected attributes of publications, such as authors, conference / journal, year, abstract, and keywords, our **analysis** focused on the identification of different categories and classifications for different maps of the outcome. The first categorization is based, on the one hand, on research type facets, such as evaluation research, solution proposal, philosophical paper, opinion paper, and experience paper (Wieringa, et al., 2006) and, on the other hand, on contribution type facets, such as model, theory, framework, guideline, lessons learned, advice, and tool (Shaw, 2003). The third common categorization aspect, the focus type facet (Paternoster, et al., 2014), which we used in the initial study (Kuhrmann, et al., 2015), turned out to be too small for our publication set. Thus, we collected the following metadata: publication vehicle, study type/method, process, and context (cf. Figure 6) (Kuhrmann, et al., 2016). As mentioned above, our focus in this chapter is on agile SPI.

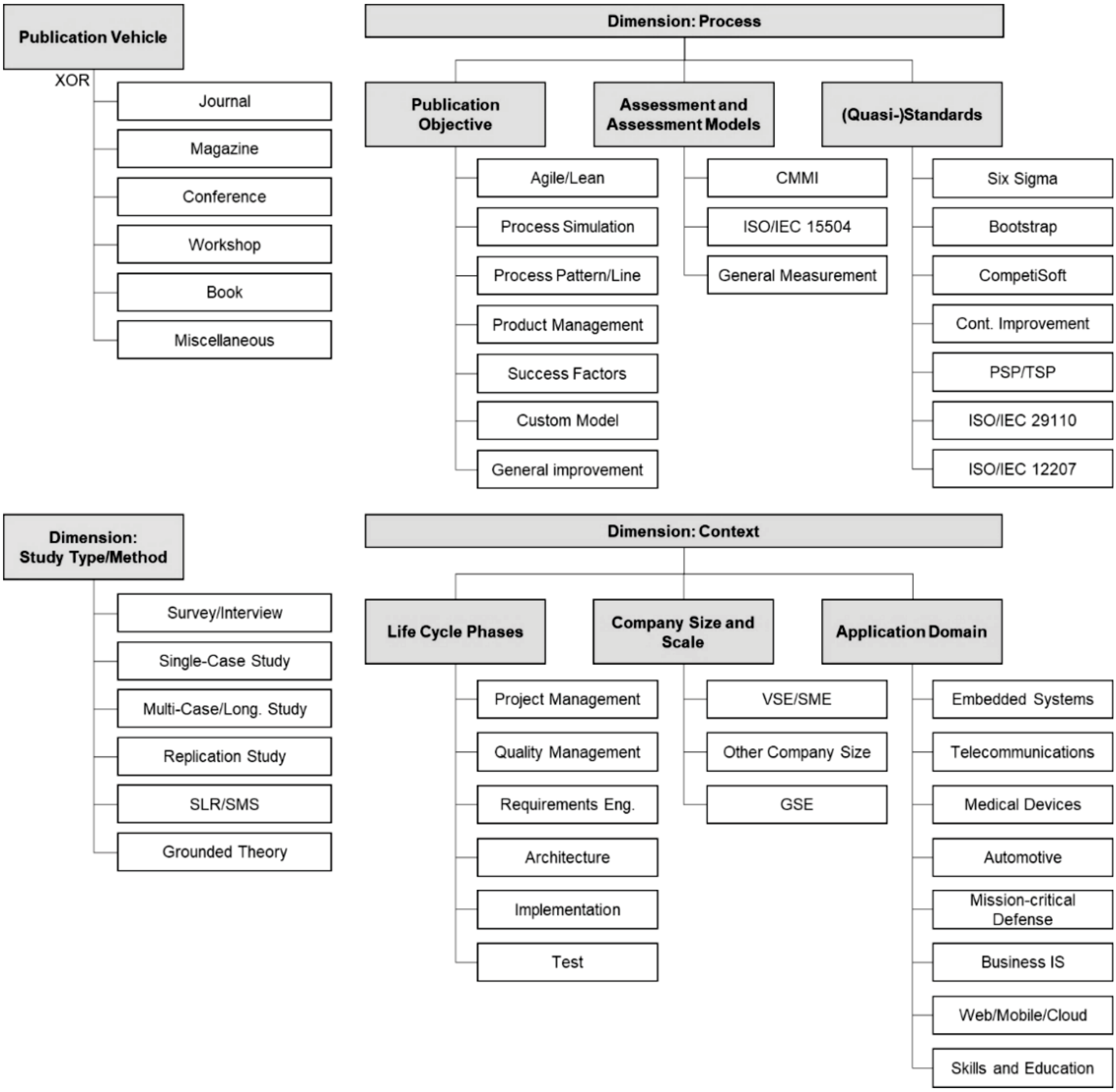


Figure 6: Collected meta-data equivalent to focus type facets (Kuhrmann, et al., 2016)

Results: In summary, the study included 769 papers on SPI published between 1989 and mid-2015. However, the main publication time started in 1996. Regarding the research type facets, the focus was on solution proposals (38%) and philosophical papers (34%). All of this data shows a clear trend in SPI towards proposing new solutions, especially considering SPI frameworks. The second trend we observed was the increasing number of papers describing lessons learned (overall as well as over time).

For the specific scope of this thesis, we will focus on the results in the process dimension with the different objectives (cf. Figure 6). Of the seven different objectives we identified, Custom Models and General Improvement were most often identified as the objective. The fourth most frequently used objective, after Success Factors, was Agile/Lean. This specific result set contained 73 papers (almost 10% of the overall data set) addressing agility in the context of SPI. Even though papers on agile development started to appear at the beginning of 1996, the "real" interest started around 2008, similar to what was found by (Salo & Abrahamsson, 2007) (cf. Figure 7).

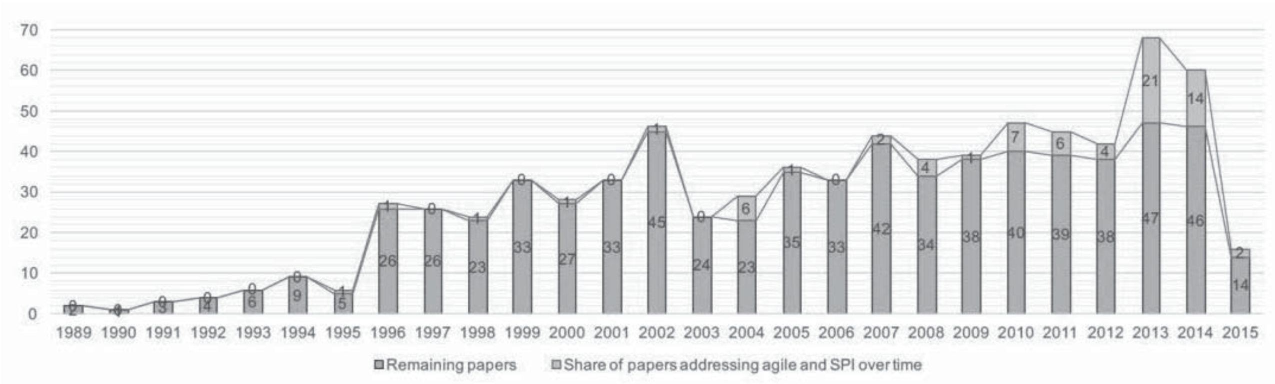


Figure 7: Agile SPI papers over time (until mid-2015) (Kuhrmann, et al., 2016)

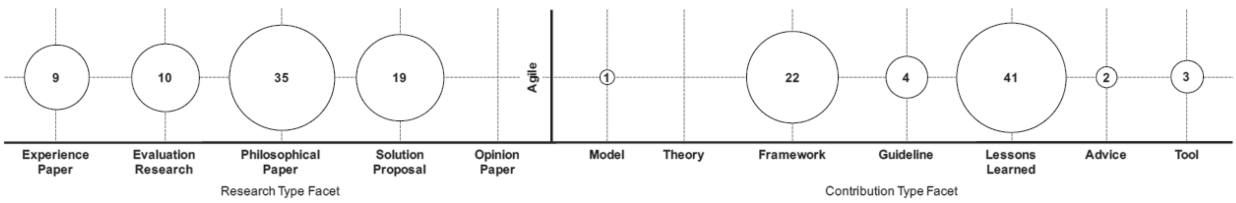


Figure 8: Overview of Agile SPI publications: research and contribution type facets

Figure 8 visualizes the research and contribution type facets of agile SPI papers. It shows balanced research among four types. Only opinion papers are missing. In contrast, the contribution type facets only focus on lessons learned and frameworks. Furthermore, if we consider more than only one aspect from the process dimension, our result set shows that most agile SPI papers deal with combinations of agility and other topics, such as standards like CMMI (Software Engineering Institute, 2010) or ISO/IEC15504 (International Organization for Standardization, 2011), or success factors (during their implementation). Nonetheless, the majority of the classified papers deals with agility as a concept for improving processes.

Since this does, on a high level, reflect exactly the idea of this thesis and the approach developed in this thesis, the next step was the refinement of this specific part of the systematic mapping study into a detailed systematic literature review including a synthesis and comparison of related state-of-the-art approaches.

3.2 Systematic Literature Review on Agile SPI

For this more in-depth study on only the literature dealing with agile SPI, we used a systematic literature review (Kitchenham, 2004) as our research method in order to be able to focus on the necessary details. Many aspects of the research design are similar to a systematic mapping study, such as search strings, databases, etc., and can be found in (Kuhrmann, et al., 2016).

Even though many aspects of the design are similar, the **research questions** to be answered are different due to the more concrete focus of this study. The detailed questions on agile SPI were:

- Which agile elements (methods and practices) are used?
- How are these agile elements used?
- What is the reason for their use?

Besides the data already extracted from the previous mapping study, we extracted information related to these three questions. This information was added to a spreadsheet.

The data analysis procedure differed from the one in the previous study (Kuhrmann, et al., 2015) because of the different scope and greater depth (e.g., full texts and not only the abstracts). First, we used the rigor-relevance model (Ivarsson & Gorschek, 2011) to identify whether proper rigor, e.g., a description of the context, was applied in a paper and whether that paper was relevant, i.e., whether its research was close to reality. These two aspects were mainly used as easy quality assurance instruments as well as for getting a general overview of the agile SPI field. Second, we classified the content of the papers according to three categories: (1) framework creation, (2) agile process optimization, and (3) problem identification. Framework creation included papers describing new methods or frameworks for agile development, with the key aspect being on the creation rather than the optimization of agile methods, which is the second category.

Because nine paper were missing or unacceptable and two were duplicates (although they had different titles and abstracts, which were used in the mapping study before), we had 62 papers in total for this detailed review.

The **results** regarding the usage of agile elements are similar to popular studies (VersionOne, 2018) (Komus & Kuberg, 2016) (Diebold & Dahlem, 2014): Even though some other agile methods as well as single agile practices were mentioned, Scrum and XP were the dominant agile methods.

We were more interested in the three categories, especially agile process optimization approaches, which are state-of-the-art approaches related to our thesis. According to these categories, 16 papers were identified as framework creation, 10 as agile process optimization, and another 12 as problem identification. The remaining papers were either written from different other single aspects or not classifiable. In all three categories, Scrum as well as XP were discussed regarding the specific topic.

Agile process optimization papers broadly covered papers that investigated already implemented cases of some agile aspect (sometimes a complete agile method) and tried to optimize its use to improve the workflow even more. The result set in this area consists of (Brown, et al., 2013)

(Esfahani, et al., 2010) (Fontana, et al., 2014) (Garzás & Paulk, 2013) (Hodgetts, 2004) (Petersen & Wohlin, 2010) (Rodriguez, et al., 2014) (Salinas, et al., 2012) (Salo & Abrahamsson, 2005) (Sato, et al., 2006).

Once we had these ten possibly related approaches, we used the rigor-relevance model of Ivarsson and Gorschek (2011) with the focus on relevance. Three of the four relevance aspects – subject, context, scale, and research method – should be fulfilled. Analyzing the resulting six papers in detail with respect to our goal of identifying related state-of-the-art approaches, we ended up with the following two: the **S**trategic pre-**A**doption **A**nalysis **F**ramework (SAAF) (Esfahani, et al., 2010) (Esfahani, 2015) and the **I**ncremental **P**rocess **A**doption (IPA) (Hodgetts, 2004). These two were the only ones that presented a repeatable approach applied in practice with enough detail. Thus, these two will be described and discussed as part of the state-of-the-art approaches in the following section.

3.3 State-of-the-Art Approaches

This section presents the state-of-the-art approaches that are related to the overall method developed within this thesis. To identify the related state-of-the-art SPI approaches, it is necessary to converge from two sides: On the one hand, we discussed earlier that agile development appears to address most of the desired improvement goals (PP1). Therefore, our first focus will be on agile approaches instead of on traditional SPI approaches. On the other hand, we identified a lack of evolutionary agile transition approaches (SP1). This was additionally confirmed by the previous studies, which found only a very small number of real agile process optimization approaches and papers. For that reason, we will also focus on evolutionary SPI approaches compared to revolutionary ones (Diebold & Zehler, 2016). The results are depicted in Figure 9, which presents these different scopes as well as related approaches organized according to these two dimensions.

In addition to this focus, we want to remind the reader that our focus in terms of context is on regulated embedded domains, such as Automotive. This was already mentioned and justified earlier. This is an important factor for deriving the requirements for the assessment of state-of-the-art approaches after their introduction.

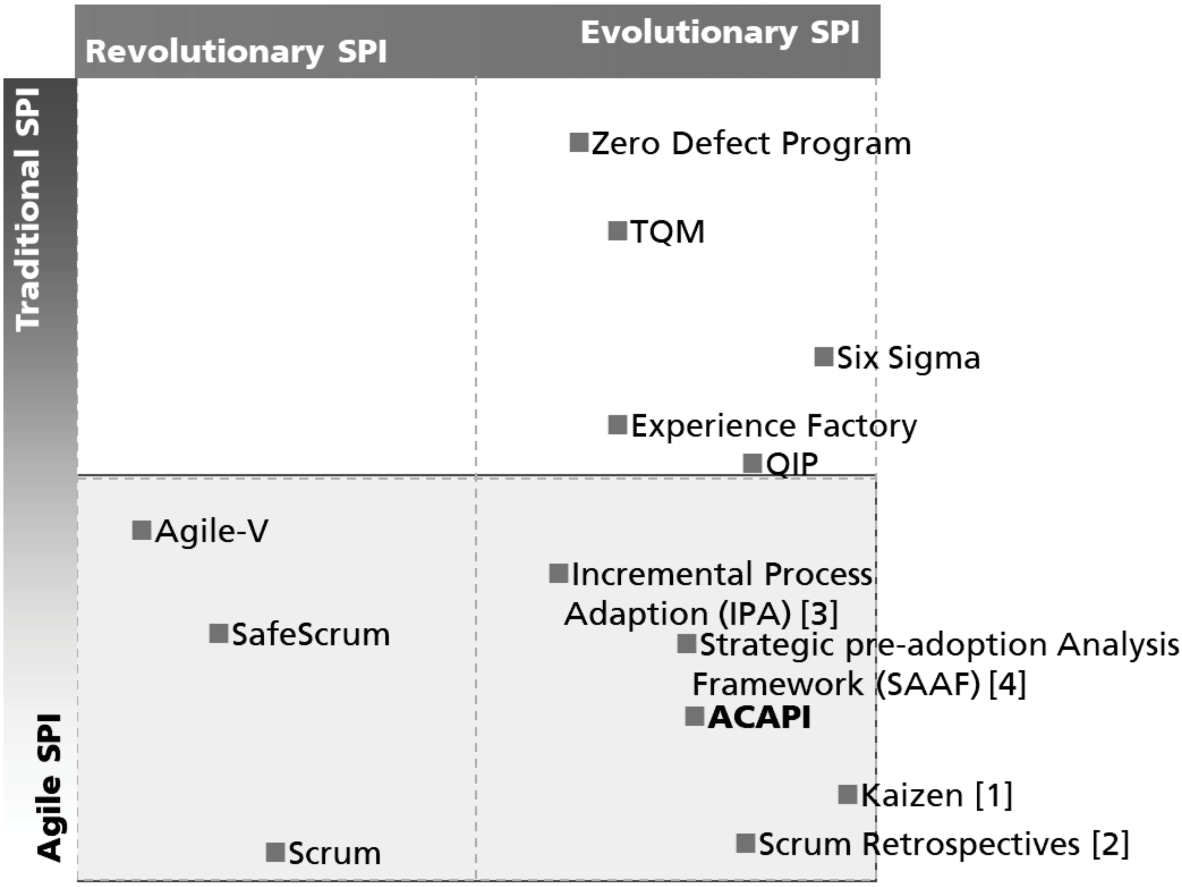


Figure 9: Past and current state-of-the-art approaches (the top-left quarter is left empty because of our focus on agile API and / or evolutionary SPI)

As stated, we focus on evolutionary agile SPI approaches (Figure 9, bottom right). Even though the focus on agile SPI is more important than that on evolutionary SPI, we are going to discuss the adjacent areas in the following subsections (Section 3.3.1 and 3.3.2). These two foci are also the reason why the top-left quarter of Figure 9 is empty. Among other approaches, the approaches that were finally selected in the mapping study presented above (cf. Section 3.2) will be covered in the discussion on evolutionary agile approaches.

3.3.1 Evolutionary Traditional SPI

Within the area of evolutionary SPI approaches, we decided to select five different approaches with short specifications as being related at least to some extent. Before explaining them in some detail, it needs to be mentioned that all of them work in an evolutionary way, of course, even if some do so more than others.

We will start with the **Zero defect program** (Office of the Assistant Secretary of Defense, 1965), which is the most traditional of the selected SPI approaches, which originates from manufacturing, and especially quality management in manufacturing. In this method, the focus is on the engineering part, with the aim being the elimination of defects (in industrial production). Today it is used more as a performance goal than as an improvement program and has been adopted in supply chains wherever large numbers of components (mainly in hardware) are produced.

The second approach, **Total Quality Management** (TQM) (George & Weimerskirch, 1998) (Houston & Dockstader, 1988), is related to our work because both approaches originate from the domain of quality management and are thus related not only indirectly with process aspects. The idea of TQM is the establishment of an organization-wide and permanent culture of continuous improvement to deliver high-quality products and services to customers. Similar to the previous approach, the invested efforts focus on developing techniques and tools for quality control. The major benefit of this approach is that many different regulatory bodies in different countries have tried to use TQM in various standards, such as the German VDI (1996).

Compared to the two previous related approaches, **Six Sigma** (Harry, 1988) is the first one that really focuses on process improvement. Although it has the same goals – improving the output of a process by identifying and removing the causes of defects and minimizing variability in business processes –, it uses a set of mainly empirical or statistical quality management methods . Each Six Sigma project in an organization follows a sequence of six steps and an up-front defined target, such as reducing costs or increasing customer satisfaction. Six Sigma is highly influenced by Deming's Plan-Do-Check-Act Cycle (Deming, 1982) and the underlying set of tools uses mainly quality management tools such as cost-benefit analysis or root cause analysis. With the focus on continuous organizational improvement started by specific initiatives (called a Six Sigma project), it is the approach with the highest evolutionary thinking in this area.

Next is the **Experience Factory** (EF) concept (Basili, et al., 2008)], which is not restricted to process improvement but started from there. Similar to Six Sigma, it is a step-wise approach focusing on specific improvement goals. The major innovation of this approach is the distinction between the project organization and the experience factory itself. The project organization's work describes the real projects, describes the context in which the projects are performed, and executes the process. This part is strongly linked with the experience factory. At the beginning, it is used for sharing data and insights from the projects to get early feedback, but at the end it also includes the project data and lessons learned for analysis and comparison. The experience factory relies on the idea of having an experience base that stores all existing experiences (which are executions in the project organization) and generalizes, adapts, and formalizes these experiences at the beginning of each new project in the organization. This and the following approach go hand in hand. However, similar to the first two, they are rather abstract approaches that do not provide detailed instructions like Six Sigma does with its set of tools.

Unlike the Experience Factory, the **Quality Improvement Paradigm** (QIP) (Basili, et al., 2008) provides a concept of how an organization can learn from project executions in detail by offering a step-wise approach. It is therefore a good complement to the EF. Within the organizational cycle, some preparatory steps are performed before the project execution is started. Similar to the Experience Factory approach, the goal(s) are set and the necessary aspects are chosen, such as process, methods, tools, etc. Within the project learning cycles, the respective aspect is executed, the results are analyzed, and the feedback is used to improve the next iteration. After the complete project execution, the overall results are analyzed and packaged such that they can be reused by other projects in the overall organizational learning cycle. This final step of packaging and storing the results and making them available to the entire organization is the link back to the Experience Factory.

To summarize the related approaches originating from the evolutionary, but more traditional SPI field, most of them focus on long-term continuous improvement of the organization. The drawback of almost of them is their origin in quality management, which means they have a concrete focus on an important part of the process lifecycle – but only on one. Furthermore, with the exception of Six Sigma, all of these concepts are quite high-level and the instructions they offer are not very concrete.

3.3.2 Revolutionary Agile SPI

Due to the predefined focus on agile SPI in this thesis, the field of revolutionary agile SPI approaches is even more important than the evolutionary approaches presented above. Figure 9 shows that the three selected approaches are all extremely revolutionary. We will first introduce the commonly adopted agile method Scrum. Since Scrum has no particular focus on embedded development, we will add two agile methods specifically for embedded domains.

Scrum (Schwaber & Sutherland, 2017) is the dominant agile method in a wide variety of different domains (VersionOne, 2018) and is commonly introduced by agile experts, such as consultants, in a big bang. The reason for this is that its authors see it as a minimal set of necessary and combined elements, roles, artifacts, and events (= meetings) to be used for developing software. With a fixed time frame of two to four weeks, each iteration begins with planning and ends with a demonstration to the customer (representative). The team meets on a daily basis for team-internal updates, e.g., regarding current impediments. Besides these basic elements of

Scrum, there is the Scrum retrospective, which takes place at the end of every iteration and is used as a reflection meeting with the complete team to identify improvement potentials. Since this is the only element in Scrum dealing with the issue of process improvement, we extracted this practice and will discuss it separately in the section on agile evolutionary SPI approaches, specifically Section 3.3.3.1. As the most frequently used agile method, Scrum is, of course, the most agile related approach of all the ones shown in Figure 9.

SafeScrum (Stålhanea, et al., 2012) (Myklebust, et al., op. 2015) is an adaptation of Scrum specifically for safety-critical software systems with very high safety function demands (up to SIL3). SafeScrum consists of common process elements from Scrum (roles, activities, and artifacts) blended with other agile practices and additional components needed to make Scrum support the development and certification of these systems. These additional components include a dedicated safety product backlog (combined or linked with the functional product backlog), activities and artifacts focusing on traceability, and a functional and RAMS[3] validation. In the initial phases, in particular, SafeScrum is being composed to match the software process lifecycle requirements of the IEC 61508 functional safety standard (International Electrotechnical Commission (IEC), 2010), which is fundamental to a wide range of domains. Its overall idea is to realize the benefits of an agile way of working and to make certification more efficient. On the one hand, this approach is less agile than Scrum because of the additional extensions and rules necessary for certification. On the other hand, similar to Scrum, the introduction of SafeScrum is intended to start in a revolutionary way, since introducing only parts of these safety elements would not satisfy regulatory bodies.

Compared to the two Scrum-related approaches, the **Agile V-Model** (Mc Hugh, et al., 2013) represents the idea of a defined hybrid approach (more traditional than the previous two). It is developed in three phases: (1) se-

[3] **RAMS** is an acronym for **R**eliability, **A**vailability, **M**aintainability, and **S**afety.

lection of underlying plan-driven software development lifecycle; (2) preparation for inclusion of agile practices into the plan-driven lifecycle; (3) identification of agile practices applicable to the development of medical device software. This shows that this approach is mainly revolutionary because these phases are only performed once and the output is fixed for a specific scope such as a team, project, company, or even the complete medical domain. Furthermore, this shows that the approach has been developed for one domain, namely medical devices, and nothing is known about its applicability in other domains. In the three phases, the final Agile V-Model includes 13 agile practices derived from a literature mapping study that mapped them to the specific stages of the common V-Model. One example is that onsite customer, customer proxy, use cases, and user stories are mapped to the Requirements Specification stage. These 13 practices are intended to overcome the perceived and actual barriers to the adoption of agile practices (McHugh, et al., 2014) in the medical domain.

3.3.3 Evolutionary Agile SPI

After providing some short information about related approaches in the neighboring fields of traditional evolutionary SPI and agile revolutionary SPI, in this section we will explain all related approaches within our focus topic on evolutionary agile SPI approaches. We will start by going into detail in the SPI-related practice of Scrum and the Kaizen approach, which is the SPI approach behind Kanban. This will be followed by two approaches that are more concrete than the others and therefore come closest to the approach presented in this thesis.

Since this is the core of this state-of-the-art chapter, each of the related work approaches will be explained in detail in its own subsection in the following.

3.3.3.1 Scrum Retrospectives

As already mentioned in the revolutionary agile SPI field (Section 3.3.2), the **Scrum Retrospective** is one of the Scrum practices involving continuous process improvement. In the Scrum Guide (Schwaber & Sutherland, 2017), it is presented as "an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next sprint". It is necessary for all team members to understand the purpose of the meeting, which is to (1) inspect how the last sprint went wrt. People, relationships, process, and tools; (2) identify and order major items that went well and potential improvements; and (3) derive a plan for implementing the identified improvements. The Scrum Master is responsible for this meeting, e.g., as the moderator and facilitator, but also as a participant as part of the overall Scrum team. Even though Scrum allows implementing improvements at any time, the retrospective provides a formal opportunity to focus on inspection and adaptation, one of the core principles behind agile development and thinking.

This general concept of a retrospective (sometimes in the past also called lessons learned) is quite an old concept – looking back at something that happened or was produced in the past. This means that retrospectives were already performed decades before agile development was invented or became famous and thus they could also be applied in traditional processes. However, in these cases retrospectives were applied after a project had been completed, if at all. With the establishment of retrospectives as a part of the overall process in every short iteration, as is the case in Scrum, retrospectives have become more powerful and are a real improvement instrument.

A retrospective and thus the Scrum Retrospective provides guidance on collecting aspects that went well and aspects that need to be improved, and is instantiated by the facilitator of the meeting. Since he or she is responsible for the output and the results of this process improvement without having any detailed instructions, this approach depends on the skills and performance of the facilitator. Even if the formal method or approach does not provide more information on the concrete instantiation, there is a lot of blog-based literature on how to instantiate a retrospective. This includes Do's and Don'ts or techniques for keeping it alive, as well as advice on how to structure the work for implementing the improvement issues, such as impediment backlogs.

3.3.3.2 Kaizen: Continuous Improvement with Kanban

Compared to the Scrum Retrospective, which is only a small part of the agile method Scrum, **Kaizen**[4] (Masaaki, 1986) (Anderson, 2010) is a continuous improvement method that uses Kanban as a complete approach. This approach relies on a culture that focuses on a working environment where all employees contribute to continuously improving quality, productivity, and customer satisfaction. Within this culture, everybody is empowered to make decisions and work on upcoming problems together. One of the major cultural values is that the management accepts mistakes, which also facilitates working with visual control mechanisms and signs. Overall, it is built upon a trustful culture that raises many social aspects, such as cooperation, respect independent of the hierarchy, or appreciation of everybody's contribution(s).

Since the visual control mechanism is one of the major aspects of Kaizen, the Kanban approach is important because it is designed to reduce the initial power of change and the resistance against it. Due to the strong connection between Kaizen, a mainly cultural approach, and Kanban, it is more important to change the organizational culture and optimize the processes instead of replacing existing processes with new ones. This is especially the case because Kanban claims that "it is better to optimize something existing, which is much easier due to change resistance" (Anderson, 2010). Small incremental changes are much easier to implement and convince people faster than a revolutionary change.

[4] from the Japanese term „kaizen", meaning "continuous improvement"

Due to the fact that everybody can start an improvement or change any time they recognize something in their team or in the larger environment, this approach is the most evolutionary one of all related work approaches represented in Figure 9. Considering the strong cultural focus in Kaizen or the Kanban approach for visualization and transparency, the agile characteristics are covered quite well. Nevertheless, these cultural issues do not consider all cultural aspects from the manifesto (Beck, et al., 2001) such that e.g. a concrete focus on organizational collaboration in working can be seen.

Since the Kaizen approach for process improvement is quite abstract (even compared to some of the abstract evolutionary traditional SPI approaches), it is important to mention some more concrete practices of Kanban as the underlying approach. These include, for example, the Work-in-Progress limit (WIP limit), which enables a “Stop-the-Line” mindset and fosters group work in solving problems.

3.3.3.3 Incremental Process Adoption (IPA)

The idea of the **Incremental Process Adoption** (IPA) by Hodgetts is that a “team would target a limited number of new practices to adopt over one or two iterations” (2004). The initial idea was motivated by Extreme Programming, where incremental adoption of XP and its practices is motivated by choosing “one practice at a time. Always addressing the most pressing problem for your team” (Beck & Andres, 2007).

This approach evolved over time through several stages, with some very successful cases and others that met with difficulties or resistance. In this approach, the team needs to choose which practices to use and adopt in the next iteration based on several factors. The major factors is the ease with which the practice can be adapted to the respective project context. Because of a team’s different levels of experience, especially regarding agility and agile practices, the idea is to look for practices that everybody considers a good idea. This can, of course, tremendously reduce the number of agile practices to be suggested. Therefore, sometimes an agile coach uses their own experiences and knowledge to suggest a practice, especially when such a practice supports others.

Even if the approach is shown to address or work implicitly with the most pressing problems, the creator of the method experienced that these problems very often do have deeper root causes. Because of this, an attempt is always made to dig down to these root causes in order to keep them in mind for the selection of the practices. This appears to be the initial difficult step before any improvement can start.

Despite these issues, the creator of this SPI approach experienced success in several teams, especially when they continuously added, modified, and sometimes removed practices over the course of several projects. This is obvious at least to some extent because when this method is applied with one or two iterations several times after each other, regardless of whether it is in one project or in similar subsequent projects, the IPA method moves more and more into the direction of the Scrum Retrospective presented above (cf. Section 3.3.3.1).

Compared to the previous and the next related evolutionary agile SPI approaches, IPA is the least agile and the least evolutionary one. This is especially the case because of the fixed improvement over one or two iterations and not more, which decreases the amount of evolution and does not make it a continuous approach. Although this also influences the rating of agility, this approach does not directly foster specific cultural aspects that are important for agile development, such as transparency.

3.3.3.4 Strategic pre-Adoption Analysis Framework (SAAF)

The **Strategic pre-Adoption Analysis Framework** (SAAF) (Esfahani, 2015) is a process improvement framework that aims at analyzing a set of agile practices prior to their enactment in an organization. Overall, this approach encompasses three components: an organizational strategic model, an evidence-based repository, and its process using the other two components.

The organizational strategic model tries to clarify the key strategic objectives of the organization and their relationship by using a strategy graph. The evidence repository provides strategic information for major agile practices. The repository with all the evidence data from systematic literature reviews as well as a few industrial experiences is no longer available at the given URL and appears to have been quite small to provide reasonable decision support for process improvement. The SAAF process starts with an initialization phase that uses the organizational strategic model. In this phase, the strategic graph is completed and the As-Is process model is reviewed. This step is followed by the strategic agile practices analysis, which analyzes candidate agile practices with respect to the strategic objectives of the previous step by means of different analysis methods, e.g., propagative strategic analysis. In the third step of the SAAF, the process model is considered for the removal of process concerns identified in the As-Is process. The creator of the SAAF himself mentions the influence of traditional SPI approaches in this step, since it is only an enrichment of the strategic graph with process concerns.

The author himself acknowledges that the small amount of evidence (because there is not that much literature about individual practices) and the small number of specific cases mean that there is a lack of adequate information for some agile practices as well as specific contexts or, even worse, a mismatch of organizational situations with the information available in the knowledge base. This is also mentioned as a major threat to the validity of the approach, as is the single instance in which it was piloted. Furthermore, some drawbacks were identified using a process model of organizations such that no further stakeholders (besides the strategic ones for the objectives) need to be considered.

Additionally, Esfahani (2015) discusses the following future work aspects that may also be relevant for the discussion and comparison of all related approaches: On the one hand, the author provides the idea of extending the SAAF as a generic strategic decision support system to increase the number of applications for strategic decision-making. On the other hand, in some scenarios, they experienced that more formalization of the complete framework would be better. However, this aspect needs to be considered carefully because it would contradict the idea of having a lightweight approach for easy access and acceptance.

Even though it does not directly consider the core values of agility, this SPI approach can be considered as an agile SPI approach. This is due to the fact that it does, on the one hand, consider agile practices as elements and, on the other hand, provides some agile nature. For example, it focuses on transparency by including information on how the practices contribute to the respective organizational objectives. Furthermore, we consider it as an evolutionary approach due to its focus on small single agile practices for improvement, even if nothing is stated on when or how often this method needs to be or is applied.

3.4 Discussion and Fulfillment of Requirements

After the detailed presentation of all the related approaches, in this section we will compare and discuss the major approaches (revolutionary and evolutionary agile SPI) based on the fulfillment of the initially defined requirements (cf. Section 2.3.2) by all these approaches.

To facilitate the comparison, Table 3 provides a high-level overview of the approaches and their assessment with respect to the different requirements and their categories. A detailed explanation of the rating will be given in the following.

Table 3: Assessment of related approaches with respect to agile SPI requirements

	Revol. SPI			Evol. SPI			
Requirements	Scrum	SafeScrum	Agile-V	Kaizen	Scrum Retro	IPA	SAAF
Goal-/problem-related SPI Req.							
REQ1: Relationship to organizational goals	-	-	o	o	-	o	+
REQ2: Stakeholder involvement	o	+	+	-	-	o	o
Evidence-based SPI Req.							
REQ3: Transparent decision-making	o	o	o	+	o	+	+
REQ4: Evidence repository	-	-	-	-	-	-	-
REQ5: Use of quantitative data	-	-	-	-	-	-	-
Continuous SPI Req.							
REQ6: Step-by-step evolution	o	-	-	+	+	o	+
REQ7: Reversibility of changes	o	o	o	+	+	+	+
REQ8: Suggesting improvement actions	o	o	o	-	-	-	+
Context Requirements							
C-REQ1: Consideration of regulatory req.	-	+	+	-	-	-	-
C-REQ2: Consideration of context	-	-	-	o	o	o	o
Maturity	+	+	o	o	+	o	-
Rigor	o	o	o	-	-	-	o

Goal- / Problem-related SPI Requirements. Considering the relationship with organizational goals, only the SAAF focuses on specific organizational improvement goals. Of the other evolutionary approaches, only Kaizen includes goals implicitly, whereas the others are not interested in them. When assessing the revolutionary approaches, only the Agile V-Model approach mentions them implicitly. The other two Scrum-related approaches are rated similar to Scrum Retrospective as their improvement mechanism.

The involvement of stakeholders is also part of goal-related SPI approaches because different stakeholders might have different problems or goals. This aspect is best considered in SafeScrum and in the Agile V-Model because there, external stakeholders are considered, even if the focus is mainly on domain and safety experts and not all others. The Kaizen approach as well as the Scrum Retrospective do not consider stakeholders other than the team. In the other approaches, this issue is left open. However, Scrum as a complete method containing the Retrospective as a major SPI aspect is a little better than the Scrum Retrospective itself because of different optional involvements within the overall process or other meetings.

Evidence-based SPI Requirements. Regarding transparent decision-making, a concrete distinction is observed between evolutionary and revolutionary approaches. All evolutionary approaches provide transparency in their decision-making. The SAAF is the one with the highest level of transparency, offering a concept for connecting agile elements with goals. The revolutionary approaches can have at least some transparency when performing and documenting the elements correctly, e.g., tracing problems to the impediment backlog items.

The requirements of the evidence repository and the qualitative consideration can be discussed together due to their completely equal assessment. No related approach, neither evolutionary nor revolutionary, uses either an evidence base or qualitative results. Regarding an evidence base, only the SAAF had the idea of using it, but never provided it. This also impacts its applicability in practice (i.e., its maturity).

Continuous SPI Requirements. For continuous improvement, it is mainly important to support step-by-step evolution (Diebold & Zehler, 2016) and not to perform a big-bang process change. The big-bang approach, also called revolutionary approach, is used in the case of Scrum, SafeScrum, and Agile V-Model, as these provide a complete (probably new) process for the development. Of these three, only Scrum might be considered a little better because of the use of the retrospective as an improvement action. Furthermore, Kaizen, the Scrum Retrospective, and the SAAF are built on step-by-step concepts. This is also the reason why all evolutionary SPI approaches presented in this related work chapter are much better in this aspect. This is similar for the requirement of reversibility of the changes because all larger changes (which are the revolutionary ones brought on by a complete method or model) are harder to reverse than smaller ones.

Finally, when discussing continuous process improvement, the question that needs to be addressed is whether concrete improvement actions are suggested by an SPI approach. Such concrete suggestions are only given by the SAAF approach, which provides agile practices as improvement actions. The other evolutionary approaches do not provide concrete actions, but rather focus on creating them based on emerging issues, e.g., as part of a retrospective. Of the revolutionary approaches, SafeScrum and the Agile V-Model at least provide some ideas on improving safety issues, which are, however, only a subset of all issues.

Contextual Requirements. Due to our specific focus on regulatory domains, the contextual requirements include the explicit consideration of regulatory requirements on the one hand and general context issues, such as team size or distribution, on the other hand. SafeScrum and the Agile V-Model explicitly support regulatory requirements. The Agile V-Model was built especially for the medical devices domain and focuses on its regulations, such as ISO62304, whereas SafeScrum tries to abstract from concrete domain-specific standards and includes safety mechanisms covering several standards. No other related approach considers this kind of requirements. The general consideration of the respective context in which a process should be established or improved is completely unimportant for all revolutionary approaches because they set up their process mainly independent of the context. This is different for the evolutionary approaches, even if they mostly do not do this explicitly.

Maturity (applicability in practice). As already shown in the Motivation (cf. Section 1.1), Scrum is the dominant agile method, which also means that it has the highest level of maturity among all approaches regarding application in practice. Due to many publications containing practical case studies, even SafeScrum is mature. The Agile V-Model as the last revolutionary approach is less mature because of fewer known practical applications. Of the evolutionary SPI approaches, only the Scrum Retrospective have reached a high level of maturity, even though they are the most frequently omitted Scrum practice (VersionOne, 2018). The SAAF is the least mature of all approaches. This is due to the fact that only one weak application has been shown and the incomplete prototype of their repository does not allow real practical application of the approach.

Rigor (abstractness vs. concrete approach). A check of the seven related approaches in terms of rigor reveals the SAAF as well as all the revolutionary approaches to have a plausible degree of detail. This is true for Scrum, SafeScrum, and the Agile V-Model because they have descriptions of what these processes should look like at the end. For example, the Scrum Guide (Schwaber & Sutherland, 2017) provides the roles, artifacts, and events (=meetings) needed to implement it. As Kaizen, Scrum Retrospectives, and the IPA are only provided in an abstract way, the SAAF is the only evolutionary approach that provides details on different steps and underlying models.

3.5 Summary

Summarizing this chapter, we started with a high-level systematic mapping study on SPI in general, providing a good overview of this field with 769 papers published between 1989 and 2015. Based on these results, we launched an SLR on agile SPI with 62 papers, resulting in the set of our related agile SPI approaches. To the identified evolutionary agile SPI approaches, we added revolutionary agile SPI approaches and other less agile evolutionary approaches. Thus, we ended up with seven related approaches: Scrum, SafeScrum, Agile V-Model, Kaizen, Scrum Retrospectives, IPA, and SAAF, which we assessed based on the necessary requirements (cf. Section 3.4).

The assessment resulted in the following ranking of the state-of-the-art approaches:

Table 4: Summary of assessed state-of-the-art approaches

Position	**Name**	**+**	**o**	**-**
1	SAAF	5	3	4
2	SafeScrum	3	4	5
2	Agile-V	2	6	4
4	Kaizen	3	3	6
4	IPA	2	5	5
6	Scrum Retrospectives	3	2	7
6	Scrum	1	6	5

Considering all the requirements that were discussed in this chapter (cf. Table 3), the SAAF is the best-rated approach. The reason for this ranking is that it fulfills five of the requirements well (relation or organizational goals, transparent decision-making, and all continuous SPI requirements) and three partly (stakeholder involvement, consideration of context issue, and rigor). With four non-fulfilled requirements, it is the best approach, along with the Agile V-Model. Even though SafeScrum can be considered to be slightly better than the Agile V-Model, these two safety-specific approaches share being the second-best assessed approaches. The next two are Kaizen and IPA, both evolutionary approaches that focus on continuous SPI requirements and continuous decision-making. Finally, Scrum and Scrum Retrospectives are assessed as the worst approaches, although they are not far behind the others considering the positive, neutral, and negative requirements. Scrum Retrospectives is quite an interesting phenomenon because, on the one hand, it fulfills as many requirements as the second-best approach, SafeScrum, but on the other hand, it has the highest number of negatively assessed requirements. This is the case because the Scrum Retrospective is the only single (agile) practice, whereas the others are complete methods, which cover more and several aspects, of course. If we keep this in mind, Scrum Retrospectives actually perform quite well. Similar to the Agile V-Model, Scrum shows a high number of neutrally rated requirements, even though it is the most mature approach.

Overall, we were able to identify that every approach has its difficulties regarding some of the defined requirements. The advantages and disadvantages of the different approaches were considered in the creation and development of the approach proposed in this thesis. Examples include the fulfillment of the continuous SPI requirements by most evolutionary SPI approaches and the stakeholder involvement and the consideration of regulatory requirements by SafeScrum and the Agile V-Model. Furthermore, we attempted to eliminate the disadvantages found in the other approaches to the greatest extent possible.

4 The ACAPI Approach

"Nothing is particularly hard
if you divide it into small jobs."
Henry Ford

This chapter contains the core of this thesis with the overall **A**pproach for goal-oriented and **C**ontext-specific **A**gile **P**rocess **I**mprovement, called ACAPI. First, the underlying model will be presented (Section 4.1), as a prerequisite for the subsequent parts of the method (Section 4.2). Then the *Agile Potential Analysis* (Section 4.3) and the *Simulation of Process Improvements* (Section 4.4) will be described. Finally, the chapter will present the limitations of the overall approach and summarize it.

4.1 Overview

The overall solution approach for the agile software process improvement presented in this thesis consists of two consecutive methods, the *Agile Potential Analysis* (Section 4.3) and the *Simulation of Process Improvements* (Section 4.4). As depicted in Figure 10, which visualizes the connection between the two sub-methods, they complement each other. Nevertheless, it is necessary to perform the *Agile Potential Analysis* first because its output serves as mandatory input for the simulation.

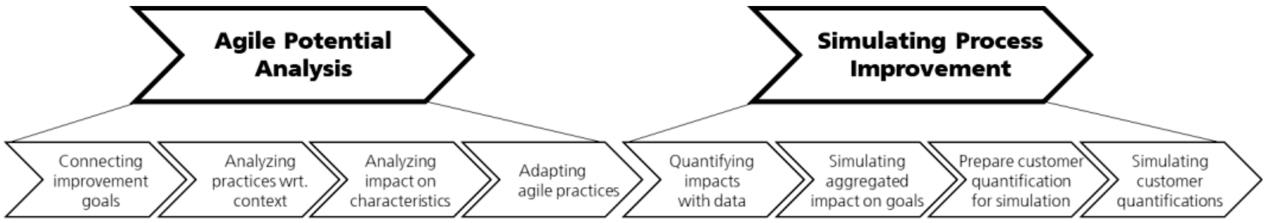


Figure 10: Connection between *Agile Potential Analysis* and *Simulation of Process Improvements*

It is important to explain the overall connection with the Experience Factory approach (Basili, et al., 2008), as it is the foundation for this entire thesis: The Agile Practice Repository (Figure 11, lower part) is a concrete instance of an experience base for process improvement using agile practices. This experience base is used by the two parts of the ACAPI approach (Figure 11, upper part), which represent the project organization of the experience factory approach. The figure shows that the overall approach uses Improvement Goals (IG_i), Context Factors (CF_i), as well as the Agile Practice Experience Base with its included Agile Practices (AP_i). Both parts of the ACAPI approach have a set of agile practices as output, integrated into the experience base with a feedback loop. Furthermore, the experience base is maintained with an improvement loop, e.g., by external evidences.

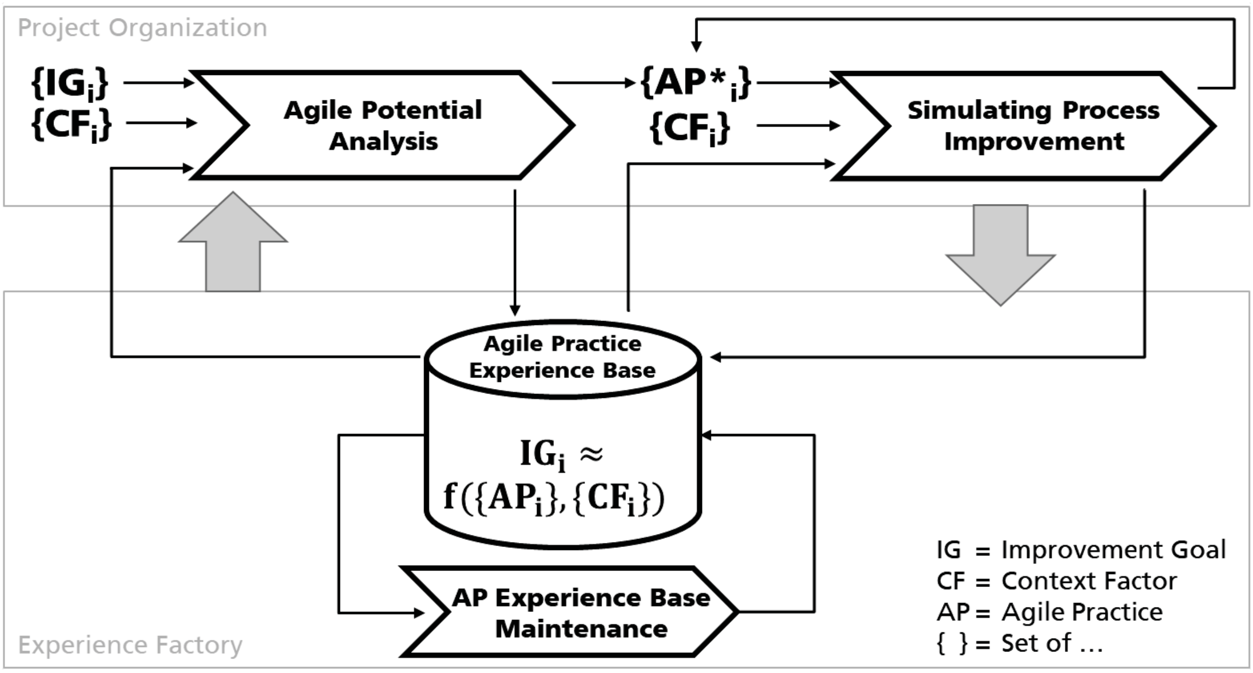


Figure 11: Experience Factory (Basili, et al., 2008) instance of the ACAPI approach

Similar to the overall experience factory, this one can be implemented for a single organization as well as across organizational boundaries. The only difference between these two scenarios is that in the first case, the experiences are only shared within the company, whereas in the second case, they are shared across company boundaries. Another difference is that in the latter case, the context is anonymized such that the company is not identifiable but context can be compared with others. In both cases, at the end of the application of ACAPI (or each part) in a team or project, the set of agile practices with the respective connection to the goals and the given context can be transferred to the experience base. Because the current data of the experience base is used when the ACAPI approach is applied, this closes the overall loop of the Experience Factory approach.

Having explained the overall connection of the two parts as well as their relationship with the underlying idea of the Experience Factory approach, the following sections will provide the details of the Agile Practice Repository (experience base, Section 4.2) as well as the two parts, the *Agile Potential Analysis* (Section 4.3) and the *Simulation of Process Improvements* (Section 4.4).

4.2 Agile Practice Repository & Impact Model

The overall ACAPI approach presented in this thesis with its two methods is built upon a necessary repository of agile practices, including an impact model of the impact of the single agile practices on different process characteristics. In the first part of this subsection, the focus will be on the schema for describing the agile practices. The second part will present the impact model. Both of these elements will be described in detail.

4.2.1 Agile Practice Description

During our initial search, performed on academic as well as non-academic literature such as blogs of agile consultants, we collected more than 300 different agile elements. However, this element list contained a lot of duplicates, with two or more different names referring to the same element, for example: "Daily StandUp", "Daily Scrum", "Stand-Up Meeting", or "Daily". While eliminating the duplicates, we also checked whether the identified practices matched our definition of agile practices (see Section 1.2 as well as (Diebold & Zehler, 2016)). Especially the second part was quite hard because we had to determine whether a practice is rather agile or traditional development, for instance in the case of reviews that started a long time before agility emerged as a topic. Finally, we decided to include everything that is used in agile development, independent of where it comes from[5]. Based on this, we ended up with a collection of about 150 agile practices. These practices currently form the elements of our repository (cf. Appendix A.1).

Besides reducing the large number of elements in the repository, it was important for us that the single agile practices are described in a common, unique, and recurring way such that the information is easy to find. In their study, Diebold et al. (2017) demonstrated the importance of a common and unique schema.

The schema itself was developed over several iterations with piloting and evaluation in between to fulfill the needs of all the different stakeholders we identified. The initial schema was published in (Diebold & Zehler, 2016). It contained eleven attributes with some further refinements; e.g., the description is refined by tasks, which can be further described as steps (similar to the Software Process Engineering Meta-Model (SPEM) (Object Management Group (OMG), 2008). A list of these initial attributes as well as their description can be found in (Diebold & Zehler, 2016). During the second iteration, a similar schema was developed together with four small and medium-sized enterprises (SMEs) in a research project[6]. Since context was most important in this project, all elements were classified according

5 This means our list might not comply with other work, such as the research on Hybrid Development Processes (HELENA) (Kuhrmann, et al., 2018), where elements were classified as well.

6 ProKoB (ProjektKontext spezifische ProzessBaustein-Orchestrierung zur Verbesserung des Entwicklungsvorgehens): www.prokob.info

to two dimensions in a matrix: One dimension is the lifecycle phase, from requirements to testing (Sommerville, 2012) (as in the other schema); the other one are the PMI process groups, from initiating to closing (Project Management Institute, 2017). In this research project, we finally specified 70 practices, but not all of them were agile practices according to our definition.

In the end, the decision was made to use all the attributes that are part of Table 5 as necessary for the repository (cf. Appendix A.2). Especially the attributes of the assessment category are important for practitioners, as they give them more in-depth information, which is helpful for assessing the different practices. This is especially important for people or companies with no or little experience in agile development, as is the case for most SMEs.

Table 5: Comparison of agile practice schemas and attributes marked for the repository

Cat.	Schema attributes	Description (partly from (Diebold & Zehler, 2016))
Description	Name	Information about what the different practices are called.
	Synonym(s) / abbreviations / translations	Information about possible other names for this agile practice. This content can be seen as a kind of alternative to the name element.
	Purpose / short description	Short description providing information about the main aim that is achieved by this practice. Should only be a few (1-2) sentences long.
	Description	Detailed description of the agile practice. In addition to the purpose (short description) that is covered, this describes how the specific practice works. Thus, the description element could be refined by a set of tasks. If there is a need to further refine such a task in more detail, a task can subsume a number of steps.
Assessment	Addressed goals	Information about improvement goals or characteristics that are addressed by using or implementing this practice.
	Advantages	In addition to the addressed goals, this aspect contains an informal description of the advantages of the agile practice.
	Disadvantages	Informal description of the disadvantages of the practice.
	Pitfalls	Short description of common pitfalls that often happen when applying the specific practice. If possible, a solution for how to avoid that pitfall is also given.
	Effort (for pilot, implementation, …)	Information about how much effort it takes to pilot, implement, or use the practice in a standard context.

Context	Contextual restriction	Information about aspects that make the application of the described practice impossible. A common examples of contextual restriction is the distribution of a team.
	Precondition	Information about aspects that need to be fulfilled before the described practice can start. The most prominent kinds of preconditions are specific work products, which are, for example, used by the practice as its input.
	Post-condition	Information about aspects that need to be fulfilled after the described practice is finished. Similar to a precondition, work products that are created or changed by the practice are examples of the output of this element.
	Variation parameter	The variation parameter describes the possibilities of changing or adapting the described practice regarding specific aspects. The most prominent example is the interval length of an iteration, which is given in the Scrum Guide [43] as 4 weeks max., but is often changed to other intervals.
	Process matrix (“high-level lifecycle phases” x “PMI process groups”)	Information about the different lifecycle processes that are addressed or covered by the agile practice. We decided to go for ISO12207, its process categories, and its processes because it is a common standard that is also used by other regulations. Information about the different process groups of PMI that are addressed or covered by the agile practice.
	Related (other) agile practices	Provides other agile practices that are related to the described one. The relationship can be of different types, e.g., from practices that are commonly combined with each other to some that exclude one or the other.
	Source / references	Defines the origin where we found the description of the practice. This may be a literature source, a website, or any other kind of source. To see how recently this practice was developed, adapted, or updated, we also consider it beneficial to provide the year this source was published.

With this schema containing these different attributes, a feasible and compartmentalized description of the single agile practices for practitioners is possible. Nevertheless, we are aware that this schema might not be enough to understand some of the more complex practices in detail. This is the reason for providing further references or sources to books or other elements. In general, this schema contains more technical aspects than cultural ones (Diebold, et al., 2015), as it was intended. Experience in practice has shown that especially for the cultural part, further consulting is needed; particularly because cultural and technical agility influence each other.

One major aspect is the positive as well as negative effects of the single agile practices covered by the addressed goals. To avoid textual and informal descriptions, the effect on commonly known process characteristics

(Diebold & Schmitt, 2016) such as transparency (of project status), customer collaboration, and employee satisfaction is covered in our repository specifically by the Agile Practice Impact Model, which forms the core of the experience factory (Basili, et al., 2008) instance.

4.2.2 Agile Practice Impact Model

The Agile Practice Impact Model (APIM) is a formal causal model for representing the influence (impact) of agile practices on different characteristics. Since it is an underlying representation, the data or information that is represented by this model is part of the overall Agile Practice Repository in the schema attribute "Addressed goals". Since it is the core of the repository and of the later analysis that is built upon this model, this subsection will provide a detailed description of the model as well as an example. (Diebold & Zehler, 2015) introduces the first high-level idea of its application in the analysis method.

Figure 12 presents the classes and relationships of the APIM meta-model as an overview.

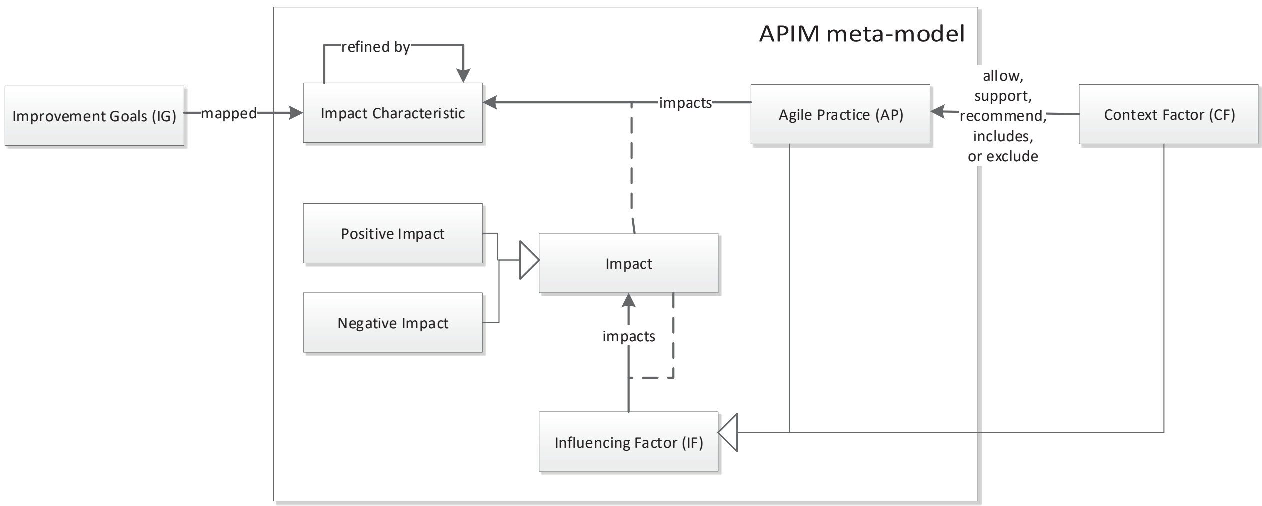


Figure 12: APIM meta-model (classes and relationships) extended by elements required for the *Agile Potential Analysis*

Classes: Agile Practices (AP = {AP_1, …, AP_x}) are "established instructions, e.g. tasks, activities, technical aspects, or guidelines, with a specific focus or aspect in the development of software which is performed according to single or less agile core values and Agile Principles" (defined in Chapter 1.2). Common examples of practices are the twelve core practices of eXtreme Programming, e.g., Pair Programming.

Definition: Impact Characteristics

Impact Characteristics (IC = {IC_1, …, IC_y}) are possible characteristics impacted by one or more agile practice. These characteristics are most often linked to (organizational) improvement goals. (Diebold & Zehler, 2015)

(Diebold & Schmitt, 2016) provide an initial list of concrete impact characteristics and a flat hierarchy of these. Furthermore, the APIM also offers the possibility of building hierarchies of impact characteristics modeled via part-of relationships. Common examples of top-level goals are customer involvement, democratization, quality, or time to market. Examples of sub-characteristics of quality could be user experience, documentation, innovation of solutions, or testability. A hierarchy is necessary because of the different abstraction levels of the characteristics that may be important. If the model contains more than one abstraction level of one of the impact characteristics, the connections from the agile practices can only lead to the lowest abstraction level. This implicitly represents an impact on the higher level(s).

Definition: Impact

An Impact represents either (1) the direct influence of an Agile Practice on Impact Characteristics or (2) the indirect influence of the influence factors on the direct connections. Each impact is represented by a "value" (not necessarily a number) and can be positive (benefit) or negative (drawback). (Diebold & Zehler, 2015)

The impact can be captured more concretely than in just these two categories. Similar to the CoBRA® method (Trendowicz, 2013), the APIM offers the possibility of specifying a single value, a range of values (with maximum and minimum), or even a probability distribution for the impact. However, even though these different types are supported by the model, they are not distinguished in the graphical representation of the APIM.

Definition: Influence Factor

Influence Factors (IF = {IF_1, …, IF_z}) represent any Context Factor that affects one or more Impacts (Diebold & Zehler, 2015).

(Clarke & O'Connor, 2012), (Kalus & Kuhrmann, 2013), and (Trendowicz & Münch, 2009) provide lists of such impact or context factors. Furthermore, examples of organizational constraints such as team size show that common lists of context factors could be used here. In addition to these factors, a single agile practice itself can also be an influence factor for other practices.

Definition: Context Factor

Context Factors (CF = {CF_1, …, CF_z}) represent any aspect or criterion that describes a (subset) of the organizational, project- or team-specific context. (Diebold & Zehler, 2015).

Relationships: The main connections of this model are unidirectional connections from the agile practices to the impact characteristics. Each connection represents the impact of exactly one agile practice on one impact characteristic (1:1-connection). In addition to this direct connection, indirect connections are also possible. These connections from an influence factor to a direct connection (from AP to IC) represent an external effect on the respective direct impact. Examples are agile practices that influence a direct impact. Since all these connections are impacts, they can be represented as either positive or negative for the specific characteristic or direct connection.

A small example of the graphical representation of the APIM is shown in Figure 13. It is an example of a completed APIM with seven agile practices that influence four high-level impact characteristics. This representation shows the relations between the different elements by placing the agile practices on the right and the impact characteristics on the left (Figure 13). The impacts between the elements are shown by the different arrows (solid for direct and dashed for indirect) and their impact orientation (positive or negative) by the sign in the circles. For example, Figure 13 shows the positive impact of code reviews on quality and the concurrent negative impact on development time.

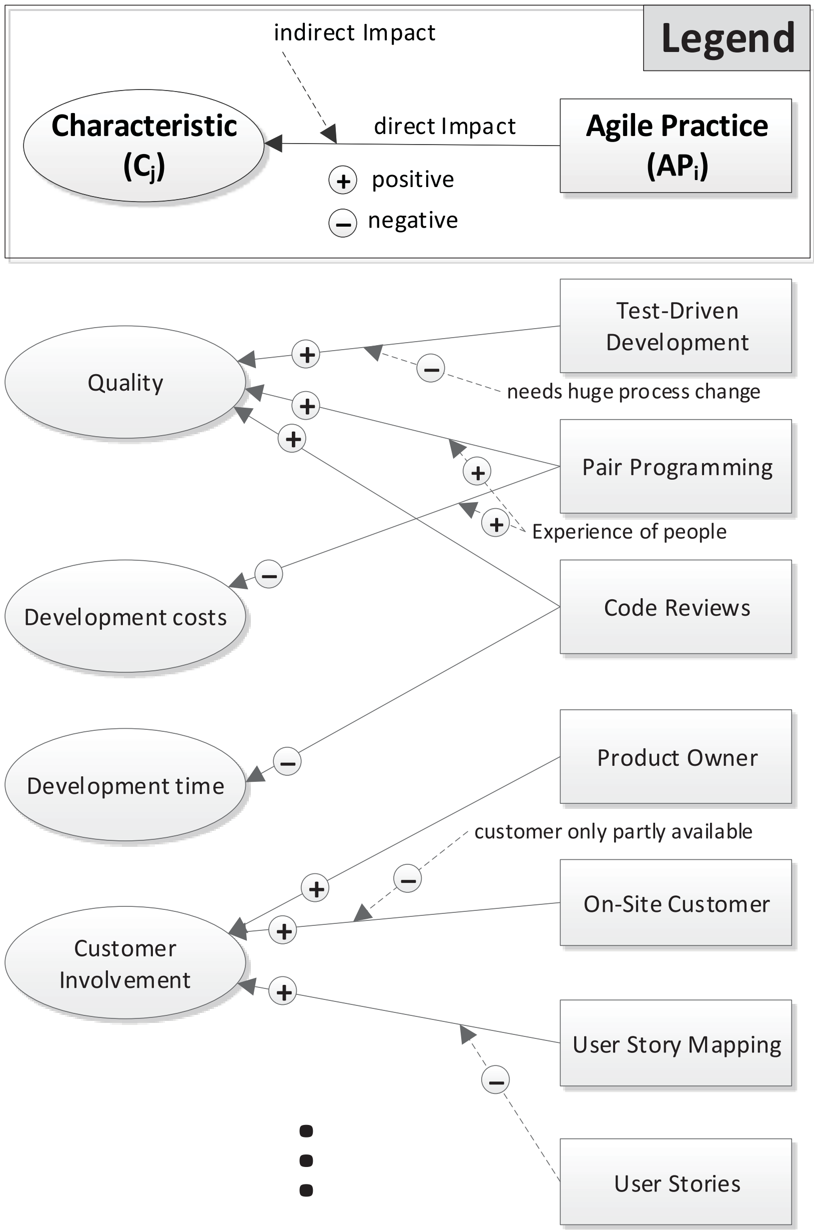


Figure 13: Graphical example of a completed APIM (Diebold & Zehler, 2015)

After knowing the graphical representation of the APIM meta-model, the main issue is getting the model filled to use it in the ACAPI approach. Two things matter in this context: (1) The impacts from the agile practices on the characteristics need to be known; (2) the effects of the influencing factors are important. They are company- or case-specific and are dealt with in the generic APIM and repository.

Filling and Maintenance Process: As already mentioned earlier, the impacts can be made explicit by different (kind of) sources, from academia or from practice.

Academic sources:

- Controlled experiments or case studies, such as (Diebold & Mayer, 2017), (Diebold, et al., 2018), (Sison & Yang, 2007), or (Diebold, et al., 2018);
- Literature studies[7] on specific individual agile practices, such as (Arisholm, et al., 2007), (McDowell, et al., 2003), (Madeyski, 2006), (Haugen, 2006), (Hulkko & Abrahamsson, 2008).

Practical sources:

- Blogs with a lot of evidence are an important source (Williams & Rainer, 2017), especially because many agile consultants share their experiences in their blogs.
- Collection of impacts with interactive posters (Diebold, et al., 2017) aimed at collecting as many expert experiences on different events as possible. This approach has so far been used for 34 events and has collected more than 3800 impacts. This large number of collected data points currently consists of 17 characteristics and 54 agile practices. They were collected on a 4-point scale (strong positive, weak positive, weak negative, strong negative).

More details on the existing data, especially on the interactive posters, can be found in Appendix A.3. Besides the mentioned sources, additional possible sources can be imagined that have not been taken into account or considered to date.

[7] Systematic Literature Reviews as well as Systematic Mapping Studies

4.3 Agile Potential Analysis

This section presents the *Agile Potential Analysis*. After an initial overview (Section 4.3.1), the different steps will be explained in detail (Sections 4.3.2 to 4.3.6). Finally, the tool support that exists for this analysis will be described (Section 4.3.7), followed by a summary (Section 4.3.8) .

4.3.1 Overview

The overview of this method encompasses the purpose, the overall input and output, as well as the high-level steps, which will be detailed later on.

Purpose: The aim of the *Agile Potential Analysis* is to support an evolutionary agile transition by identifying the potential of extending the current development process with single agile practices or practices necessary for using agile elements in the respective context.

Input: To achieve this purpose, the *Agile Potential Analysis* needs two different inputs: It is necessary to know (1) the current problems and improvement goals of the concrete projects being focusing on as well as some organizational ones, and (2) the context being considered by the analysis method, which also includes the current development process.

Output: The *Agile Potential Analysis* produces a customized project-specific transition backlog, including a prioritized set of agile practices as Transition Backlog Items (TBI). Even if the main intention of the analysis is the initial filling of the transition backlog for an agile transition, this can be performed iteratively and used for reprioritizing or filling this backlog during the transition.

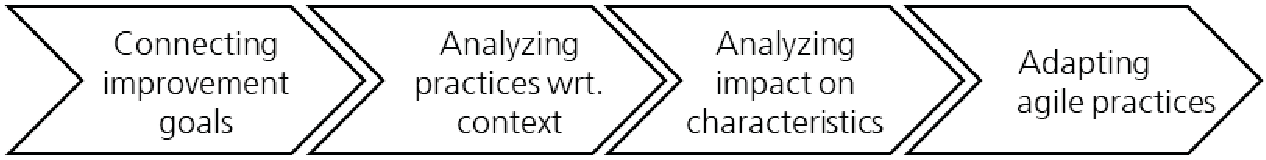


Figure 14: *Agile Potential Analysis* steps

The *Agile Potential Analysis* consists of several steps. Figure 14 shows that the method includes the initial connection with improvement goals (Step 1) and consideration of the context (Step2). This is used to analyze the impact (Step 3) and decide on possible adaptations of agile practices (Step 4). Each step will be described in detail in the subsequent sections, including its objective, inputs, outputs, activities, and possible tool support.

4.3.2 Step 1: Connecting Improvement Goals

As presented in the state of the practice (Section 2.3), software process improvement initiatives need to be goal-oriented, especially in the context of agile where evangelists drive many aspects, such as their own new methods. Therefore, including and working with improvement goals is very important for the *Agile Potential Analysis*. In its first step, the objective is to identify current problems and align them with possible improvement goals in order to guide the selection of agile practices.

Table 6 summarizes the most important elements of this step. A detailed description of each activity comprising the "Connecting Improvement Goals" step will be provided below.

Table 6: *Agile Potential Analysis*: Step 1 - Connecting Improvement Goals

Step 1: Connecting Improvement Goals	
Objective	The objective of this step is to identify current problems and align them with possible improvement goals in order to guide the appropriate selection of agile practices.
Inputs	• Agile Practice Impact Model
Activities	1. Identify and prioritize improvement goals 2. Connect improvement goals with process characteristics 3. Deselect unimportant agile practices 4. <optional> Mark agile practices already in use 5. Prioritize agile practices
Tools	• Tools for eliciting and documenting organizational and project-specific improvement goals, e.g., GQM+Strategies. Example tools include standard office tools. • Tools for modeling the connection of the improvement goals with APIM. Example tools are either spreadsheet applications or modeling tools (such as EA).
Outputs	• List of prioritized improvement goals • List of agile practices already in use • List of prioritized agile practices for process improvement

Activity 1.1: In the "identify and prioritize improvement goals" activity, we discuss with one or several stakeholders current development or organizational problems. These are collected and prioritized either in a workshop with all stakeholders or by conducting interviews with them. During the activity, it is not mandatory to prioritize the goals, but the more detailed the input is, the more precise the output. What the output of this activity might look like is illustrated in Figure 15 (left part).

Activity 1.2: The objective of the next activity, "Connect improvement goals with process characteristics", is to map the identified and prioritized improvement goals to the process characteristics of the APIM (or its data basis). In the ideal case, there is a 1:1 mapping between the goals and the characteristics, with every mentioned improvement goal being connected with the model (cf. Figure 15). However, in reality, it often happens that not all problems or improvement goals can be mapped due to various reasons, such as organizational issues of a complete program. Furthermore, the mapping is normally an n:m relationship because one problem might be connected with several characteristics and more than one problem may relate to the same characteristic. By prioritizing the problems, it is also possible to prioritize the APIM process characteristics using the following algorithm:

```
Connect_Goals_Characteristics()

for (i=1 to #characteristics)

   for (j=1 to #problems)

      if (problem_j is mapped to characteristic_i)
         characteristic_value = characteristic_value + problem_value;
```

Assuming that higher priority is represented by a higher value, for every process characteristic the values of all the mapped improvement goals or problems are summed up. Based on this algorithm, it is now also possible to prioritize the characteristics.

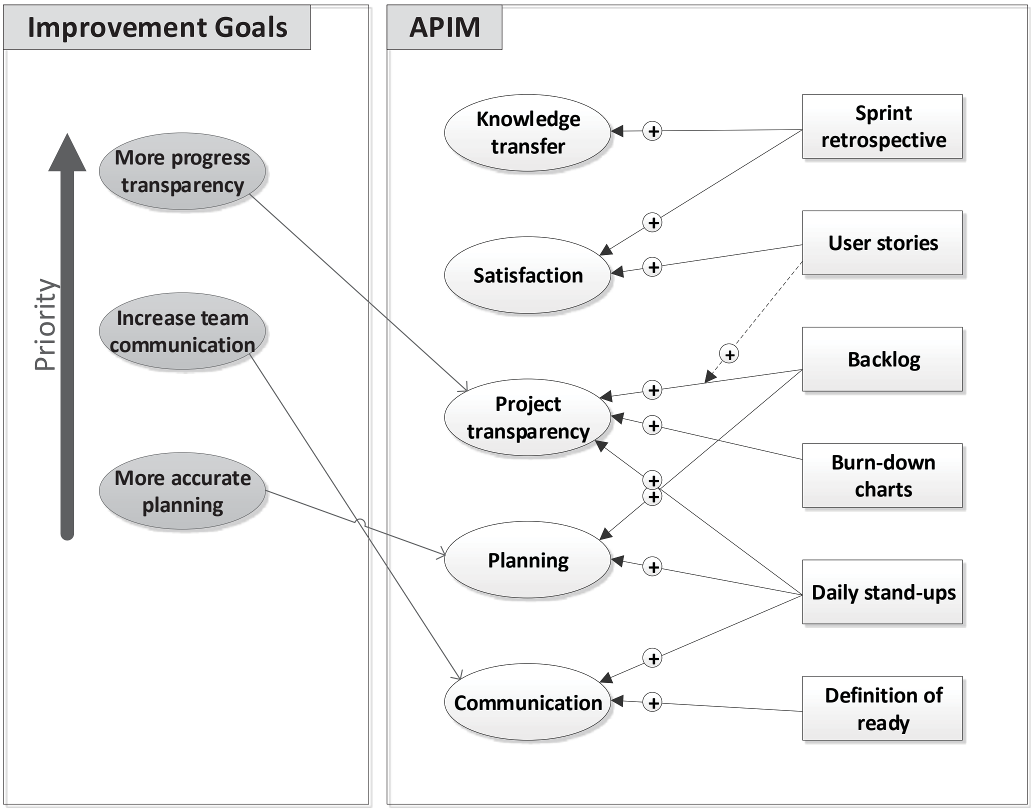


Figure 15: Example outcome after Step 1 - Activity 2 of the *Agile Potential Analysis*

Activity 1.3: In the activity "Deselect unimportant agile practices" the APIM and its data are used for the first time. With information about the impacts, all the practices that do not have any impact on the selected and prioritized characteristics can be eliminated from the individual instance of the model. In this case, the complexity reduction of the model depends on the number of the selected prioritized characteristics.

Activity 1.4: The activity "Mark agile practices in use" is an optional one in this step (but needs to be performed the latest at the beginning of Step 2). Nevertheless, in this activity the objective is to further reduce the number of possible agile practices by marking agile practices already in use (cf. Figure 16, Sprint Retrospective). This should be done because the *Agile Potential Analysis* should not suggest practices as improvement that people are already using. During the first applications of the analysis (cf. Case Studies in Chapter 5.1), we experienced that sometimes it would be better not to delete practices already in use, but rather mark them differently. Even if this makes the model more complex and thus influences the following steps, these practices are presented in a similar way as the suggested ones. The advantage of this is to flag these marked practices or their application for questioning.

Activity 1.5: Independent of whether or how the practices in use are marked or eliminated, the final activity of Step 1 is "Prioritize agile practices". This is performed using the prioritized characteristics as well as all the existing APIM data (cf. Figure 16; top practices mean highest priority). Before using the data for the algorithm, it is necessary to decide whether to use all the data or only the data of the impact connections of a practice with a characteristic that reaches a defined threshold. Taking the latest data of Section 4.2.2, we are currently applying the threshold of more than five expert evidences for each connection[8]. Another aspect that needs to be considered during all prioritizations of agile practices (which also applies to all subsequent AP prioritization activities) is the fact that some practices are preconditions for others, meaning that precondition practices need to have higher priority or that, if a precondition is excluded, other practices are also excluded as a consequence.

[8] The value for the threshold might vary or increase over time when more data is added to our experience base.

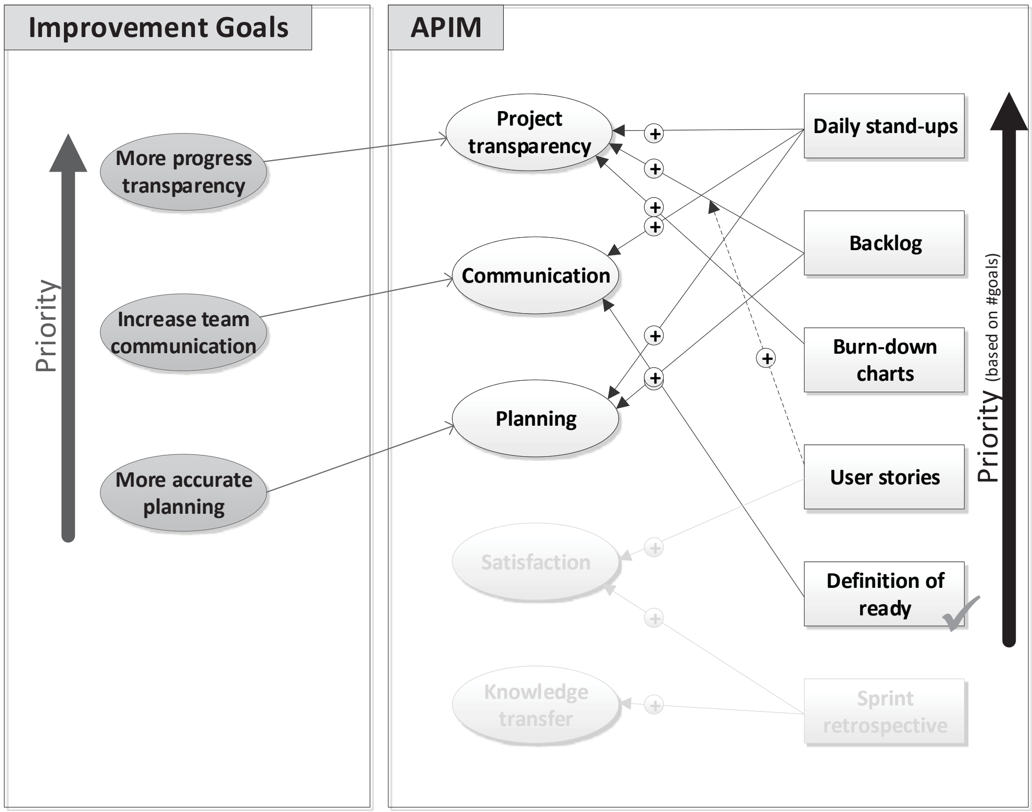


Figure 16: Example outcome after Step 1 - Activity 5 of the *Agile Potential Analysis*

In the description of the algorithm, traditional sorting algorithms are used in some parts because we do have numbers for comparison. In a simplified way the algorithm looks as follows:

```
Sort_AgilePractices(AP[], NumberOfGoals or Value)

   ig[] := sortedImprovementGoalsList;
   for (i=1 to ig[].length)

      ap[] := select all AP[] addressing ig[i];

      if (NumberofGoals)
         result[] := result[] + Sort_AgilePracticesBy_NumberOfGoals(ap[]);

      else
         result[] := results[] + Sort_AgilePractices_ByValue(ap[]);

      ap[] := select all AP[] not addressing ig[i];
      i++;
```

```
Sort_AgilePracticesBy_X(AP[])

    Select top_ap[i]
    if (several ap[i] are the same)
       Sort_AgilePracticesBy_X (ap[] with same value)

    else
       result[] := result[] + top_ap[i]
                   + Sort_AgilePracticesBy_X(ap[] without top_ap[i])
```

4.3.3 Step 2: Analyzing Practices with respect to the Context

Step 2 takes into account the respective context issues. In our case, context does not only consider aspects like organizational issues or team size and distribution, but also covers regulatory constraints and other aspects. We initially used (Kalus & Kuhrmann, 2013) and (Clarke & O'Connor, 2012) as starting points for characterizing the context. Furthermore, from our point of view, the current software or system development process is also an important part of the context and needs to be considered in this and the following step.

Table 7 summarizes the most important elements of this step. We will provide a detailed description of each activity comprising the "Analyzing Practices with respect to the Context" step below.

Table 7: *Agile Potential Analysis*: Step 2 – Analyzing agile practices wrt. context

Step 2: Analyzing Practices with respect to the Context	
Objective	The objective of this step is to identify current context factors that are constraining the exclusion of agile practices.
Inputs	• List of prioritized agile practices for process improvement (output of Step 1)
Activities	1. Mark agile practices already in use <if not done in Step 1> 2. Identify contextual factors 3. Connect contextual factors directly with agile practices 4. Deselect excluded agile practices 5. (Re-)Prioritize agile practices
Tools	• Tools for eliciting and documenting organizational and project-specific context information. • Tools for modeling the connection of context with APIM. Example tools are MS Excel or EA.
Outputs	• List of appropriately prioritized agile practices for process improvement • List of contextual constraints

Activity 2.1: The activity “Mark agile practices already in use”, which was an optional activity in step 1, is necessary the latest at the beginning of this step. The idea is to further reduce the number of suggested agile practices. Furthermore, the possible variations of this activity explained above also apply for this one as well. Since the practices already in use are a part of the current development process, this aspect fits even better in this step covering the context.

Activity 2.2: The activity “Identify contextual factors” consists of identifying current problems and improvement goals. The objective here is to collect as many contextual factors as possible, independent of whether they are used as constraints or not. Constraints are specific contextual factors that directly affect the practices, like an exclusion. They are not part of the core APIM (cf. Section 4.2.2). Such context information can best be elicited in a workshop or by conducting interviews. It is also possible to combine both forms of elicitation since not all the activities of all steps need to be performed subsequently. Seeing that it is sometimes hard to collect contextual information, we found it quite helpful to work with a kind of questionnaire or existing references giving indications what might be common or helpful contextual information for our analysis method. For this purpose, we mainly used (Kalus & Kuhrmann, 2013) as supporting material.

Since we are focusing on regulated domains, where agile is not so common yet, the different regulations are one of the most important pieces of contextual information to collect. As an example, the Automotive domain as the strongest regulated embedded domain in Germany is restricted by several regulations, mainly Automotive SPICE™ (VDA QMC Working Group 13, 2015) as the process standard as well as ISO26262 (International Organization for Standardization, 2011) for safety. Although more regulations exist in this or in other domains, we will illustrate the necessary steps by using one of them as an example.

Activity 2.3: In the activity "Connect contextual factors directly with agile practices", the objective is to find the direct connections from the context constraints to the remaining practices. Connection is a high-level term, and the most important connection types include, but are not limited to: *allow*, *support*, *recommend*, or *include* as positive connections as well as *exclude* as a negative connection (Diebold & Zehler, 2015) (cf. Figure 17). Especially excluding connections are important for the next activity, whereas positive connections are used for further (re-)prioritization.

Together with some industry partners, mainly SMEs, we identified the following factors as common constraints (Table 8), using the list of (Kalus & Kuhrmann, 2013):

Table 8: Context factors as constraints and possible scales

Context Factors	Scale
Team size	one, two to five, six to ten, larger than ten
Team distribution	onsite, onshore, nearshore, or offshore
Project duration	0-4 weeks, 1-3 month(s), 3-6 months, more than 6 months
Customer and user availability	none, daily, weekly, monthly

Based on the scales for these constraints, 41 agile practices were evaluated as to whether they needed to be excluded. This resulted in 21 exclusion connections. With this information in the repository, the practices can be excluded automatically when selecting the context factor on the defined scale. In the future, this experience base of existing exclusions or connections should grow after every application of the *Agile Potential Analysis*, because in normal applications a lot more context constraints are identified and need to be connected manually by companies and agile experts during the analysis. These manual connections can be reused later on as suggestions, similar to our research data.

If a regulation like Automotive SPICE was identified, not the complete regulation is linked with the individual agile practice. In such a case, the single regulatory requirements of the standard are connected, e.g., work products or base practices in Automotive SPICE. Since Automotive SPICE (which specifies "What" to do) and agile approaches (which specify "How" to do it) are not contradictory, the connection is a supporting connection (Diebold, et al., 2017). Diebold and Richter (2017) mapped 155 agile practices to the 185 Automotive SPICE requirements, resulting in 772 supporting connections. This means that if this regulation becomes an influencing factor, all these supporting connections can be included in the model and be used in the subsequent activities.

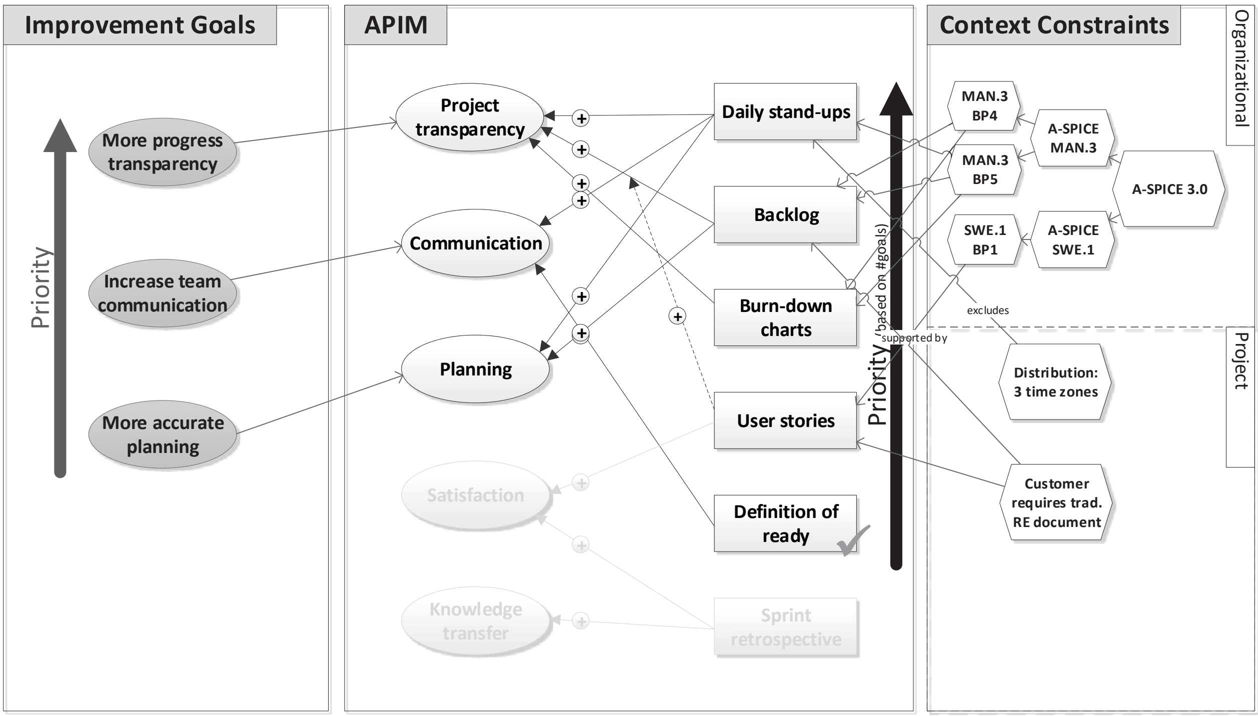


Figure 17: Example outcome after Step 2 - Activity 3 of the *Agile Potential Analysis*

Activity 2.4: In the "Deselect excluded agile practices" activity, all agile practices that are connected, e.g., via an exclude or similar connection, are excluded. This leads to another reduction of the list of practices as well as the overall complexity of the model (cf. Figure 18).

Activity 2.5: Finally, in Step 2, the activity "(Re-)Prioritize agile practices" uses the new information, such as excluded practices or supporting connections, and again (cf. Step 1 – Activity 5) evaluates the importance of the priority of the practices (cf. Figure 18). Automation is possible for all the practices that are not excluded in this step. However, for the positive connections, especially since the context space might have many different manifestations, this prioritization is most often done manually. One exception might be the given example of Automotive SPICE, where the number of Automotive SPICE requirements supported by one agile practice could be included as a parameter in the algorithm presented above.

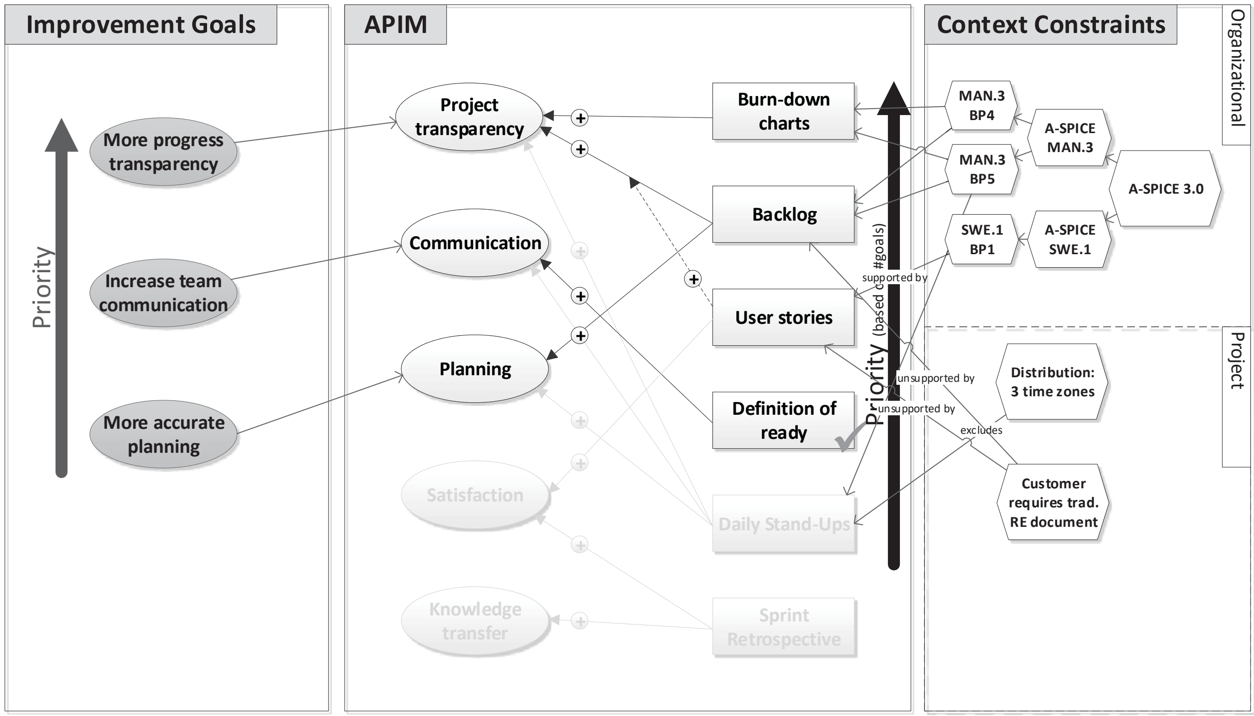


Figure 18: Example outcome after Step 2 - Activity 5 of the *Agile Potential Analysis*

4.3.4 Step 3: Analyzing Impact on Process Characteristics

The reason for having this step is the same as the previous step, i.e., consideration of the context for the individual SPI initiative. Nevertheless, the focus is a different one than in Step 2. Here, the objective is to identify which contextual factors positively or negatively influence the impact of agile practices on process characteristics.

Table 9 summarizes the most important elements of this step. We will provide a detailed description of each activity comprising the "Analyzing Impacts on Process Characteristics" step below.

Table 9: *Agile Potential Analysis*: Step 3 - Analyzing Impact on Process Characteristics

Step 3: Analyzing Impact on Process Characteristics	
Objective	The objective of this step is to identify current individual context factors including, e.g., the current processes that have a positive or negative influence on the impacts of agile practices.
Inputs	• List of prioritized agile practices for process Improvement • List of contextual constraints (both outputs of Step 2)
Activities	1. Identify more contextual factors 2. Connect contextual factors with impacts 3. (Re-)Evaluate impacts 4. (Re-)Prioritize agile practices
Tools	• Tools for weighting all connections with APIM and prioritizing the agile practices. An example tool is MS Excel.
Outputs	• List of prioritized agile practices for process Improvement • List of context-influencing factors and their influences (positive vs. negative)

Activity 3.1: Step 3 begins with a collection and identification activity: "Identify more contextual factors". Compared to the "Identify contextual factors" activity of the step before, it works in exactly the same way and all the contextual information can also be collected together using different methods, such as the specified workshops or interviews. A comparison of all context factors with the more constraining ones may yield some new aspects.

Activity 3.2: It is necessary to correlate the identified context information with the existing impacts. For this reason, all context factors are considered and evaluated as to whether they influence the single impacts of the agile practices. An important aspect in this activity is that an agile practice itself might be an influencing factor for some impacts. For example, the impact of Story Mapping as a practice is different depending on whether or not User stories are used to document the requirements. This also means that agile practices are one of the central elements of the model, as they might be some kind of precondition for other practices (which is similar to a context factor but not necessarily an excluding one, resulting in a reprioritization) or a factor affecting other impacts.

Because of the large amount of possible contextual information, this activity is mainly performed manually. We started collecting some data for this activity with the SMEs for usage and automation. In contrast to exclusion, it was more complex because we marked those practices that have at least one impact that might have an influence. Since we only found that context factors never impact positively and negatively at the same time. Although this activity as well as the given data is similar to the connection of the context constraints, storing the data and learning from the data is much more difficult due to the individual nature of the factors.

Activity 3.3: Because Activity 3.2 might have changed the impacts, the objective of "(Re-)Evaluate impacts" is the detailed consideration and further evaluation of these impacts. This does not necessarily mean that the impacts are quantified as the given impacts of the repository or the APIM, but they are at least re-evaluated according to the new and given influencing context factors. This evaluation is especially complex if more than one influencing factor is involved per impact. Since the context factors might be company-specific and an individual aggregation would be necessary for every case, we decided that a manual expert evaluation is the most appropriate way.

Activity 3.4: Finally, in Step 3, the activity "(Re-)Prioritize agile practices" is performed in a similar way as the last activity of Steps 1 and 2, but it cannot be automated. Because of this, the algorithm provided above cannot be used either. This means that an expert's evaluation is required of how or whether these context factors, or better yet the (re-)evaluated impacts (not necessarily given as values), result in re-prioritization of the agile practices.

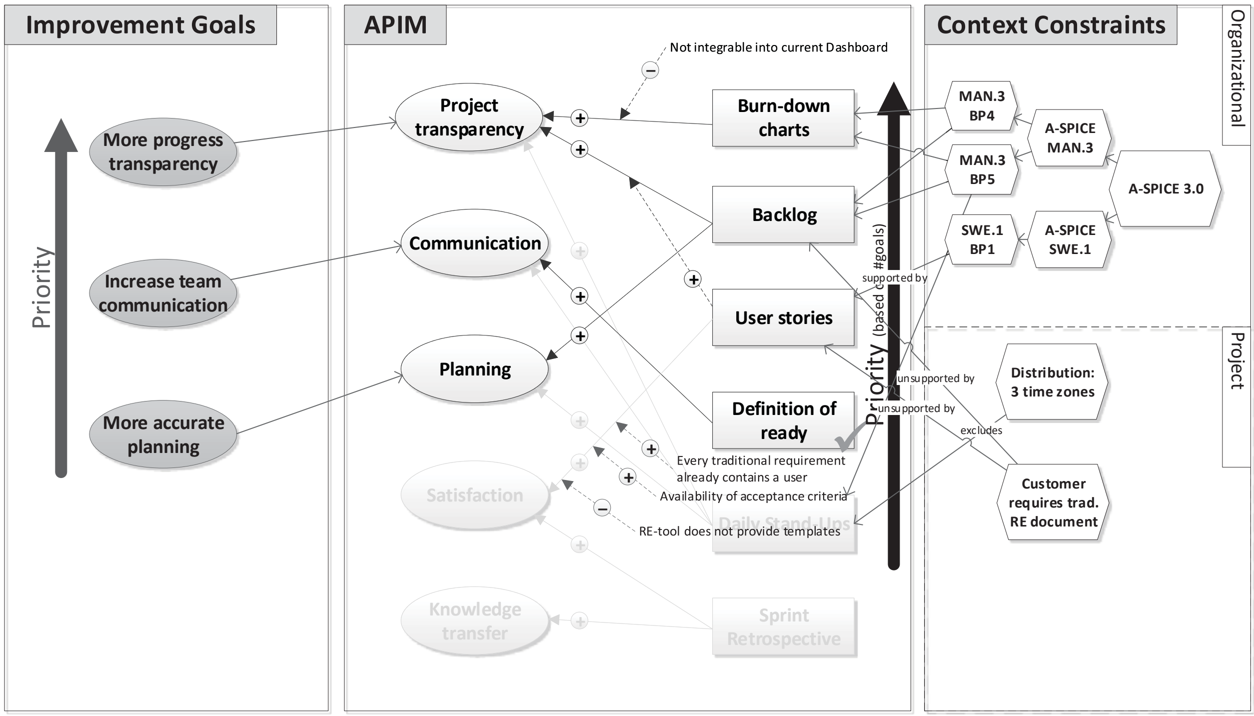


Figure 19: Example outcome after Step 3 - Activity 4 of the *Agile Potential Analysis*

4.3.5 Step 4: Adapting Agile Practices

The final step of the *Agile Potential Analysis* deals with the adaptation of agile practices to the individual context before their introduction. The need for this comes from the adaptations of complete agile methods (Diebold, et al., 2015), e.g., the so-called ScrumButs (Elorantaa, et al., 2016). Having collected all the context factors, using them for identifying, checking, and evaluating possible agile practice adaptations is the logical next step.

Table 10 summarizes the most important elements of this step. We will provide a detailed description of each activity comprising the "Adapting Agile Practices" step below.

Table 10: *Agile Potential Analysis*: Step 4 - Adapting Agile Practices

Step 4: Adapting Agile Practices	
Objective	The objective of this step is to identify and evaluate context-specific adaptations of the set of agile practices.
Inputs	• List of prioritized agile practices for process improvement (output of Step 3) • Agile Practice Repository containing the commonly used agile practices described in a schema, e.g., containing variation parameters
Activities	1. Suggest agile practices adaptations 2. Evaluate suitability of agile practices adaptations 3. (Re-)Evaluate impacts 4. (Re-)Prioritize agile practices
Tools	• Tools for describing or specifying the schema, incl. variations / adaptations of the agile practices. Example tools include MS Word, MS Excel, more technical tools like XML, or database formats. • The suggestion could be supported by EA if it was used for the previous steps and includes all schema information (or at least the variations).
Outputs	• Adaptations for every (remaining) agile practice (if possible or appropriate)

Activity 4.1: The "Suggest agile practice adaptations" activity is built upon the Agile Practice Repository (cf. Section 4.2) and mainly uses the "variation parameter" attribute. In this activity, the analysis method provides possible variations or adaptations of the practices to the users, especially based on the given contextual information. For example, the given variation parameters of the practice Daily Stand-ups are the duration of the meeting, which is normally a maximum of 15 minutes, and the frequency, which is normally every day. A context factor like team size influences the duration of the meeting since every participant should answer the three questions posed. In general, this activity is automated in part due to the given variation parameters of the agile practices and a few concrete implemented contextual factors, such as team size.

Activity 4.2: Besides the automated aspect of the selection, the next activity "Evaluate suitability of agile practices adaptations" is necessary to select appropriate adaptations or variations of the practices. This is more or less the manual equivalent to the automated activity before and needs to be performed by bringing together experts from the agile domain who know the practices and their variations in detail with experts from the company who have enough contextual background. Thus, the outcome of this activity are adaptations of those practices from the previous list that have the highest priority.

Activity 4.3: Now that the adapted practices are available, in the activity "(Re-)Evaluate impacts" the objective is to scrutinize the existing impacts to determine whether they are the same as without adaptation or whether a change is necessary. For example, conducting the Daily Stand-up only every other day does not affect the increasing transparency from our point of view but might reduce the positive impact on productivity due to longer timespans for resolving blockers.

Activity 4.4: The activity "(Re-)Prioritize agile practices" works similar to the prioritization activity of the agile practices (Step 3 – Activity 4). If the previous activities of this step did not change any information used for the prioritization, such as the impact information, this activity would not be necessary. Normally, if adaptations are meaningful, it is necessary to re-consider and probably re-prioritize.

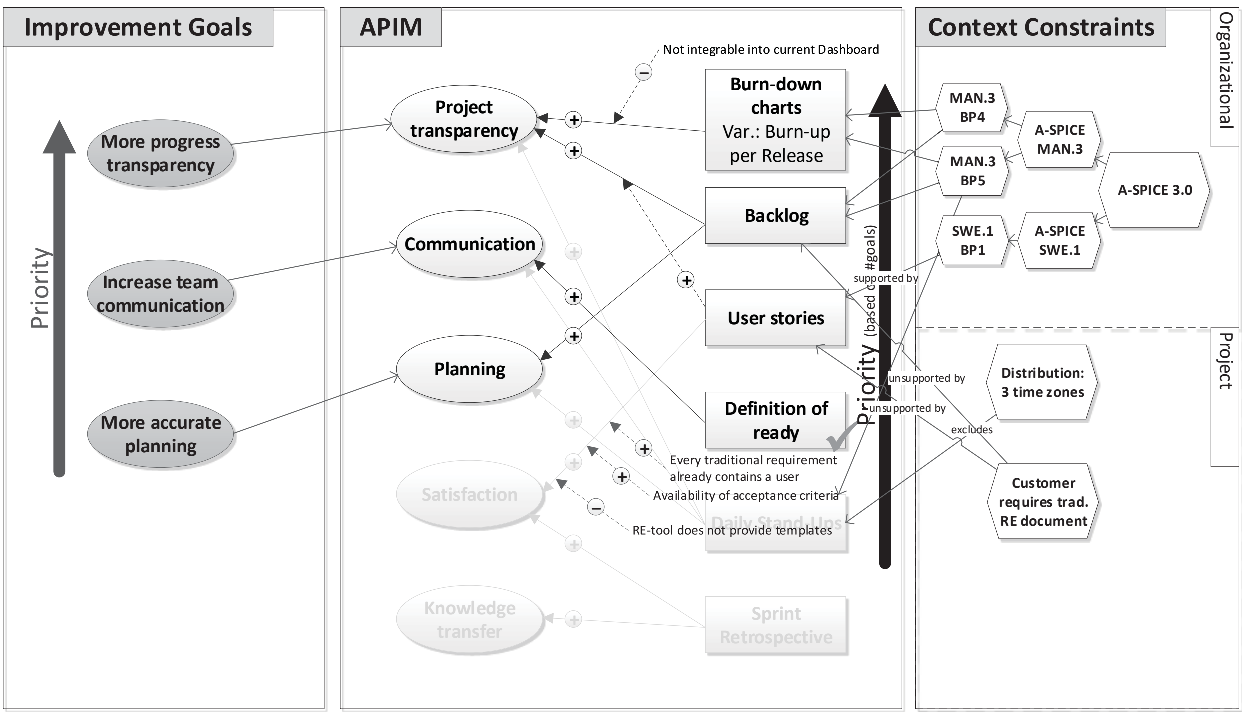


Figure 20: Example outcome after Step 4 - Activity 4 of the *Agile Potential Analysis*

4.3.6 Final Outcome

After all four steps of the *Agile Potential Analysis* have been performed with their detailed activities, the outcome is a context-specific, prioritized list of agile practices. As briefly described in the overview, we call this prioritized list the transition backlog.

Each agile practice in the backlog is a Transition Backlog Item (TBI). Besides an ID and the name of the practice itself, the backlog also contains the impact values of the practices on the selected and prioritized characteristics (cf. Table 11). This is necessary and helpful as an explanation for the people who were not involved during the performance of the *Agile Potential Analysis* but are now confronted with the results. Furthermore, the backlog can, of course, be extended with further information found in the repository, such as risk factors or required effort.

Table 11: Example of a transition backlog (with prioritized characteristics from left to right and some example fake data)

ID	**Agile Practice Name**	**Risk**	**Transparency**	**Knowledge Transfer**	**…**
AP1	Daily Stand-up	Low	1,42	1,32	…
AP2	Test-driven Development	High	0,82	0,96	…
AP3	Pair Programming	Low	-	1,4	…
…	…	…	…	…	…

Besides this listing or tabular representation of the transition backlog, it would also be possible to present it in a graphical representation by using the APIM as well as the necessary external connections with the model. However, as the complexity of the model representation rapidly increases even if there is only a small number of practices and characteristics (seen in the figures above), we normally use the tabular representation. Furthermore, the listing representation is also easier to manage in a tool, such as JIRA, confluence, or even Excel.

4.3.7 Tool Support

Tool support for the *Agile Potential Analysis* is two-fold: On the one hand, an Enterprise Architect (EA) plugin supports the modeling and visual representation, e.g. for the APIM. On the other hand, Excel tooling provides automation and facilitates working with the *Agile Potential Analysis*.

EA Plugin. The purpose of the EA plugin is the modeling aspect of the analysis with a focus on the APIM. For this reason, the plugin contains all the modeling elements and connections presented in Section 4.2.2. In addition, it has further attributes for possible usage in the *Agile Potential Analysis*, such as the tagging of agile practices already in use to enable filtering. The power of EA as a modeling tool for our purpose is, on the one hand, its visual representation, which is easier to understand for most people. On the other hand, features such as filtering or reduction are helpful in the use of the *Agile Potential Analysis*. Nevertheless, the visual representation also has its disadvantages, i.e., the increasing complexity with the rising number of elements (e.g., agile practices, improvement goals, context constraints or factors).

Excel Tooling. The tool support in MS Excel contains some templates that can be used to fill in the necessary information as well as some Visual Basic scripts that are packaged in an Excel add-in. The Excel tooling mainly supports Steps 1 – 3 of the analysis method, and we also have ideas to extend it to Step 4. A template for filling in the individual problems or improvement goals supports the first step. Since the mapping needs to be done manually, a script only supports the sorting and prioritization of goals and process characteristics. In the transition to Step 2, a script supports the selection and prioritization of the agile practices. This is based on the characteristics and the impact data (cf. Section 4.2.2), where the current data can be easily integrated. This output is then included in another template for mapping context constraints and factors (Steps 2 and 3). Furthermore, the Excel add-in supports many smaller tasks or steps in the analysis, e.g., insertion or export of new data from APIM, prioritization of various things such as goals, process characteristics, or practices, as well as integration of common constraints or factors such as team size, etc. The presentation of the final output of the *Agile Potential Analysis* shows the prioritized agile practices, the context influencing these, as well as their average impact, so that all necessary information is shown together. Even if this view is sometimes hard to understand and reduces some information, it contains most of the transition backlog items (the suggested agile practices).

Due to the easier implementation and the detailed knowledge of the information and the analysis method, we are currently using Excel tooling as support for the *Agile Potential Analysis* and for further extending this approach. Furthermore, most companies use MS Excel and can therefore directly use, edit, and work with the results, whereas EA is mostly found only in larger companies.

4.3.8 Summary

Summarizing the *Agile Potential Analysis* including all its necessary foundations as well as its tool support, we see the need to perform all four of the defined and described steps. It might be possible to leave out the third step but this would have consequences for the final priorities of the transition backlog containing the agile practices. Figure 21 visualizes the complete method in depth, with the activities of all previous steps as well as their different inputs and outputs.

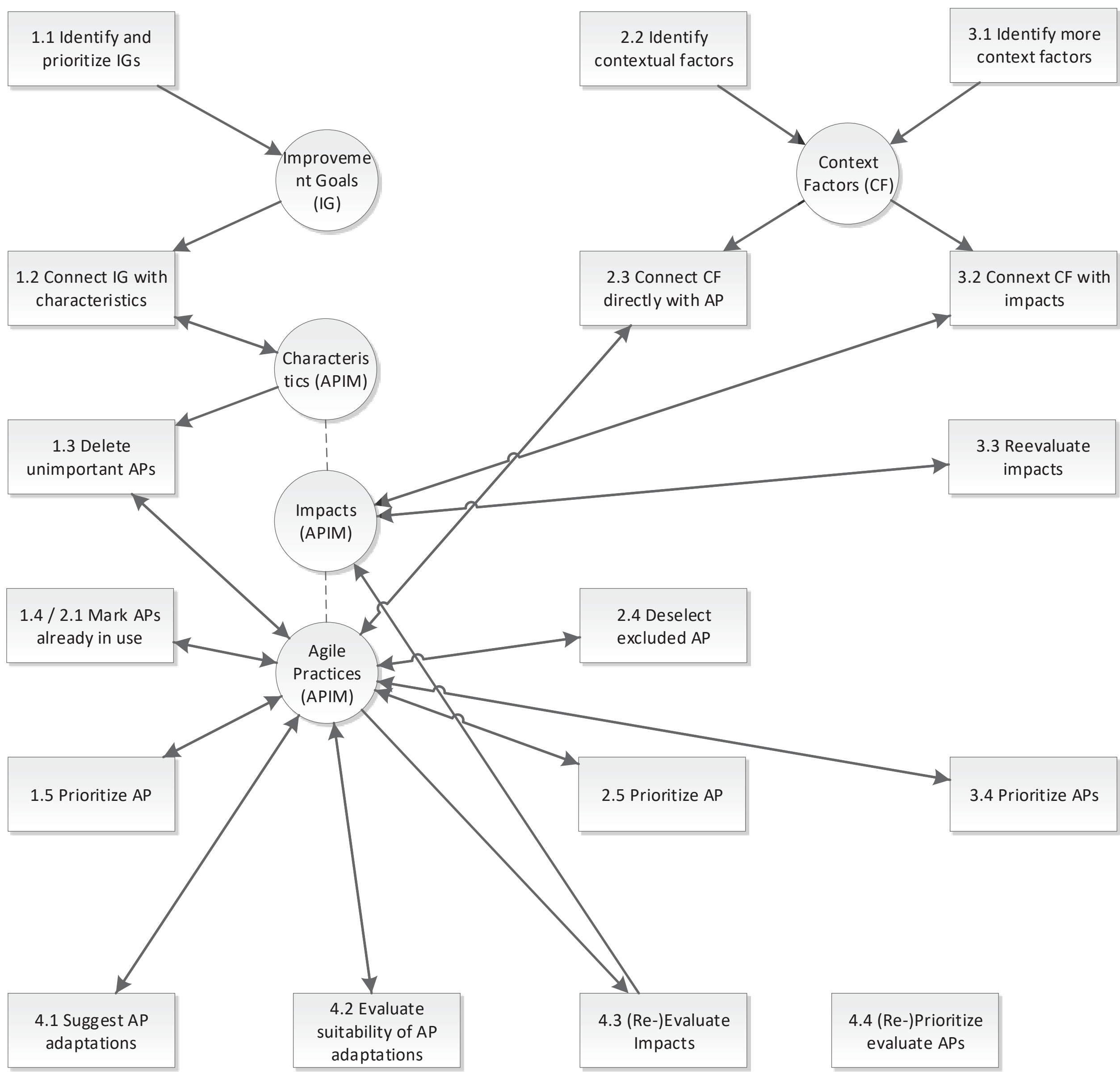


Figure 21: Detailed Illustration of the sub-steps of the *Agile Potential Analysis* including inputs and outputs (Step 1 on the left, Step 2 in the middle, Step 3 on the right; Step 4 at the bottom)

This figure illustrating the incoming and outgoing artifacts shows that the core of this method are the agile practices in combination with the impacts, both of which are included in the APIM. The agile practices are used by most steps. Usually the list of agile practices is not only used, but changed, for instance with regard to prioritization or adaptation. Except for the usage of the agile practices artifacts, the different steps do not intersect with each other because the other artifacts they are using are normally used on their own, e.g., goals or problems in Step 1, context constraints in Step 2, and context factors in Step 3. The only possible intersection that might occur could be a piece of contextual information being a constraint as well as factor.

Despite the complexity of the *Agile Potential Analysis*, as depicted in Figure 21 with the large number of activities working with the different artifacts, it can be easily applied by agile experts, especially if they have all the existing data of the APIM. From the effort perspective, the company in which the analysis method is going to be applied only needs to invest effort for the initial identification, collection, and prioritization of the relevant information. This can normally be kept to less than 2.5 person-days depending on the details. From the viewpoint of the analysis executer, more effort is necessary for performing all the different steps and activities in detail.

4.4 Simulation of Process Improvement

This section presents the *Simulation of Process Improvements*, which is the extension of the *Agile Potential Analysis* and thus the second part of the method developed and presented in this thesis. After an initial overview (Section 4.4.1), the different steps will be explained in detail (Sections 4.4.2 to 4.4.5. In Section 4.4.6, the existing tool support will be described, before this part concludes with a summary (Section 4.4.7).

4.4.1 Overview

The overview of this method encompasses the purpose, overall input and output, as well as the high-level steps, which will be detailed later on.

Purpose: The aim of the *Simulation of Process Improvements* is to further support the agile transition by objectifying the decision-making of the *Agile Potential Analysis*. This can be done by performing a mathematical and simulation-based analysis of different variables, such as selection or deselection of agile practices.

Input: To achieve this purpose, the *Simulation of Process Improvements* needs (1) the tailored APIM model, (2) existing impact data, e.g., the collected impacts from the interactive posters, and (3) experts from the case company. These experts should know both the agile practices and the company context.

Output: The *Simulation of Process Improvements* mainly produces a visual representation of the impacts of the agile practices as well as their individual contribution to the different improvement goals for possible re-prioritization or confirmation of the transition backlog. Furthermore, due to the usage of more and quantitative data, the decision to take agile improvement actions should be more objective and transparent.

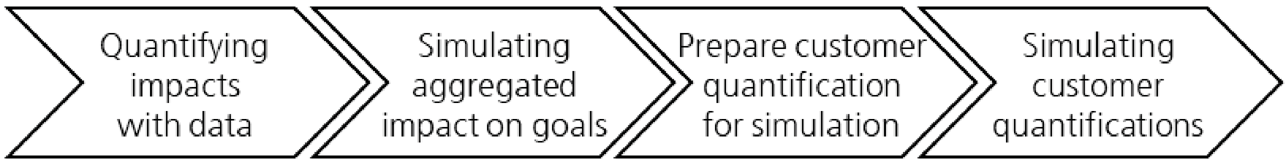


Figure 22: *Simulation of Process Improvement s*teps

Figure 22 shows the different steps. The first simulation uses the existing impact data from experts to come up with some aggregated impact on the improvement goals (Steps 1 & 2). The second simulation (Steps 3 & 4) can be applied if a user is willing to provide their own quantification (which is used similar to the CoBRA® simulation (Trendowicz, 2013)). Each step will described in detail in the subsequent sections, including its objective, inputs, outputs, activities, and possible tool support.

4.4.2 Step 1: Quantifying Impacts with Existing Data

Due to the fact that most often quantification is used to improve decision support, and since it was mentioned as a requirement in the state of the practice (Section 2.3), we use the impact data collected from experts for visualization purposes. This is the initial step towards supporting better decision- making when it comes to selecting appropriate agile practices.

Table 12 summarizes the most important elements of this step.

Table 12: *Simulation of Process Improvements*: Step 1 – Quantifying Impacts with Existing Data

Step 1: Quantifying Impacts with Existing Data	
Objective	The objective of this step is to use the impact data collected from experts for an initial visualization of this quantitative data to support decision-making.
Inputs	• Existing impacts of APIM, e.g. from interactive poster collection
Activities	1. Get impact data collected from experts. 2. Transform frequency data into hit probabilities. 3. Calculate confidence intervals for hit probabilities. 4. Visualize hit probabilities with their respective lower and upper confidence bounds.
Tools	• Tools for transferring and working with the collected impact data. An example tool is MS Excel. • Tools for calculating statistical values and analyses, in this case the confidence intervals. An example tool is R.
Outputs	• List of impacts (strong negative, negative, positive, strong positive) of agile practices on different process characteristics o Hit probability for each impact category o Lower and upper confidence bounds for the hit probabilities • Visualization of the three values

Activity 1.1: The activity "Get impact data collected from experts" is needed to export the existing data from the repository and more specifically the APIM database. The data are exported manually into a processable data format, such as a spreadsheet. The next activities need to have the data for every category (strong negative, negative, positive, strong positive) in every impact connection individually. In this activity, it would also be possible to further filter the data and only select a subset, e.g., data from a specific conference or domain such as Automotive.

Activity 1.2: Once the frequencies for every category are available following Activity 1.1, "Transform data into hit probabilities" is a complete mathematical transformation of the data. For every impact connection, the hit probability, i.e., the relative share, for each category (from strong negative to strong positive) is calculated.

Activity 1.3: The activity "Calculate confidence interval" (upper and lower confidence bounds) addresses the statistical uncertainty of the derived hit probability. The data set of every impact connection is a realization drawn from a multinomial distribution. For this reason, simultaneous confidence intervals (CI) for multinomial proportions are calculated for the consideration of uncertainty (Glaz & Sison, 1999).

There are different methods for calculating simultaneous CIs for multinomial data. Six of these methods, including the Bonferroni-adjusted formula of Goodman, are discussed in (May & Johnson, 1997) with concrete examples that can be transferred to our scenario. In (May & Johnson, 1997), May and Johnson used simulations to test the coverage probability of each of these formulas. They conclude that the simple Bonferroni-adjusted formula of Goodman (1965) "performs well in most practical situations when the number of categories is greater than 2 and each cell count is greater than 5, provided the number of categories is not too large" (May & Johnson, 1997). This would fit for most parts of our data collected so far. Furthermore, 5 is the commonly selected threshold in the *Agile Potential Analysis* (Step 1 - Activity 5). Furthermore, the methods that use sample variance, Fitzpatrick and Scott (1986) and Quesenberry and Hurst (1964), are "poor" according to (May & Johnson, 1997). The remaining methods "perform reasonably well with respect to coverage probability but are often too wide" (May & Johnson, 1997). A nice feature of the Goodman methods and some other methods is that they do not necessarily produce symmetrical intervals, but are always within the interval [0,1]. For these different reasons, we decided to use the Goodman method with a confidence level of 1 - 0.05, meaning that the true, but unknown value is within the interval with a probability of 95%.

Since this step does not necessarily depend on the tailored APIM, we are using all the extracted data and calculate the CIs with their upper (U_CB) and lower confidence bounds (L_CB) using the above-mentioned method. We are currently using R[9] as the tool for calculating the confidence bounds.

[9] https://www.r-project.org/about.html

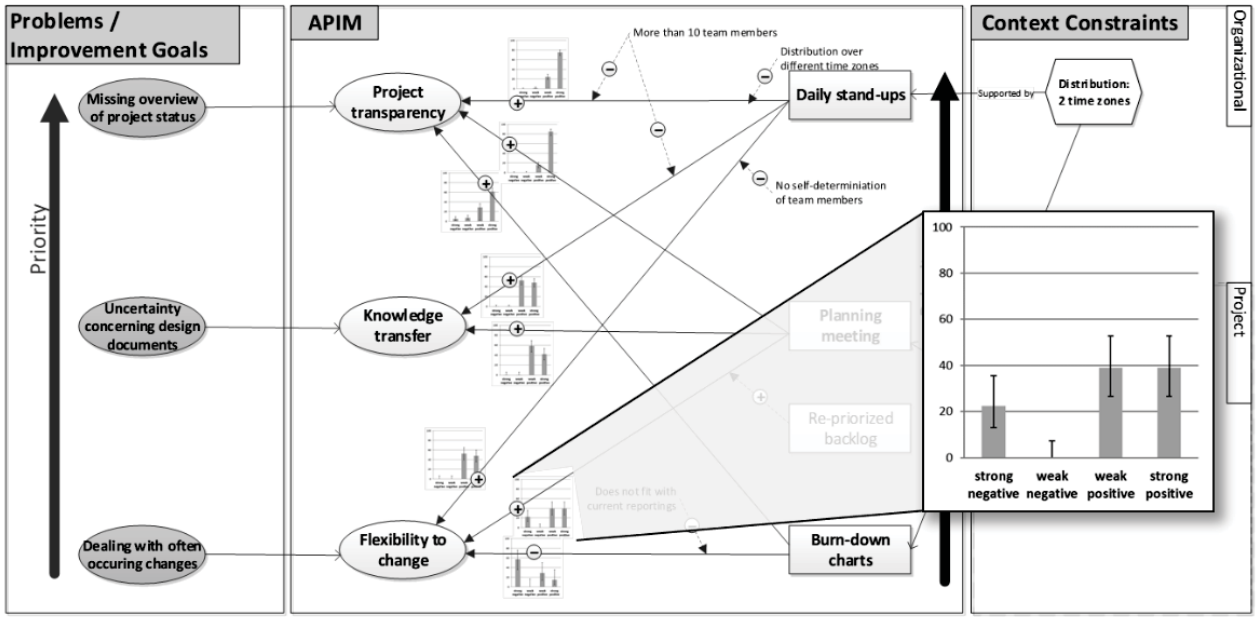


Figure 23: Example outcome after Step 1 - Activity 4 of the *Simulation of Process Improvements*

Activity 1.4: In the activity "Visualize hit probabilities with lower and upper confidence bounds", the tailored APIM model from the *Agile Potential Analysis* is combined with the processed data from the previous activity. Figure 23 illustrates the calculated probability as well as the CI (Figure 23, black bar) in one visualization per impact connection. The example shows eight impact visualizations, with a zoomed excerpt of the impact of the Daily Meeting on the characteristic "Flexibility to change".

4.4.3 Step 2: Simulating Aggregated Impacts on Goals / Characteristics

As presented in the overview of the simulation, this step now uses the data of the previous step for the creation of the aggregated impacts on the improvement goals. This is an aggregation step aimed at reducing complexity. It is not necessary to have a visualization for each impact. One visualization for each goal or characteristic would be enough.

Table 13 summarizes the most important elements of this step. We will provide a detailed description of each activity comprising the "Simulating Aggregated Impacts on Goals/Characteristics" step below.

Table 13: *Simulation of Process Improvements*: Step 2 – Simulating Aggregated Impacts on Goals/Characteristics

Step 2: Simulating Aggregated Impacts on Goals/Characteristics	
Objective	The objective of this step is to combine the existing quantified impacts for each improvement goal based on their individually tailored Agile Practice Impact Model from the *Agile Potential Analysis*.
Inputs	• Hit probabilities for agreement categories with the respective lower and upper confidence bounds for all impacts • Tailored APIM (because not all impact connections might be relevant in this step)
Activities	1. Derive (probability) distribution of improvement goals by means of Monte-Carlo simulation 2. Visualize impacts for each improvement goal 3. Derive worst- and best-case scenario 4. Calculate agile practice contribution 5. Visualize agile practice contribution
Tools	• Tools for generating randomized values to perform simulation(s) on this data. Example tools for simulation and calculation are MS Excel or R.
Outputs	• Aggregated impacts on improvement goals (incl. their visualization) • Maximal and minimal impact contribution of the single agile practices for the individual improvement goals • List of prioritized agile practices for process improvement → Transition backlog

Activity 2.1: The activity "Derive (probability) distribution of improvement goals by means of Monte-Carlo simulation" is the first one that does not deal with a single impact but aggregates the impacts for each of the process characteristics. For this step, we chose to run a Monte-Carlo simulation (Doubilet, et al., 1985), which works as follows: For a given process characteristic, we simulate random numbers of each of the agile practice impacts on the process characteristic and aggregate them into a distribution of their weighted[10] sum.

In concrete terms, for each impact i (from an agile practice to a given process characteristic), the hit probability of each of the four categories **s**trong **n**egative (SN), **w**eak **n**egative (WN), **w**eak **p**ositive (WP), and **s**trong **p**ositive (SP), denoted by $\hat{\theta}^i = \left(\hat{p}^i_{SN}, \hat{p}^i_{WN}, \hat{p}^i_{WP}, \hat{p}^i_{SP}\right)$, and the total number of answers collected n^i, i.e., $n^i = n^i_{SN} + n^i_{WN} + n^i_{WP} + n^i_{SP}$, where, e.g., n^i_{SP} denotes the total number of hits in the SP category (for the given process characteristic) are used as parameters of a multinomial distribution $Mult(n^i, \hat{\theta}^i)$. For each agile practice that has an impact i on a

[10] Currently we are using equal weighting, which could also be adapted.

specific process characteristic, we simulate realizations of $Mult(n^i, \hat{\theta}^i)$-distributed random variables (e.g., in Figure 24 and Figure 25 the three incoming edges to project transparency, where the total number of simulations should be equal among all impact connections).

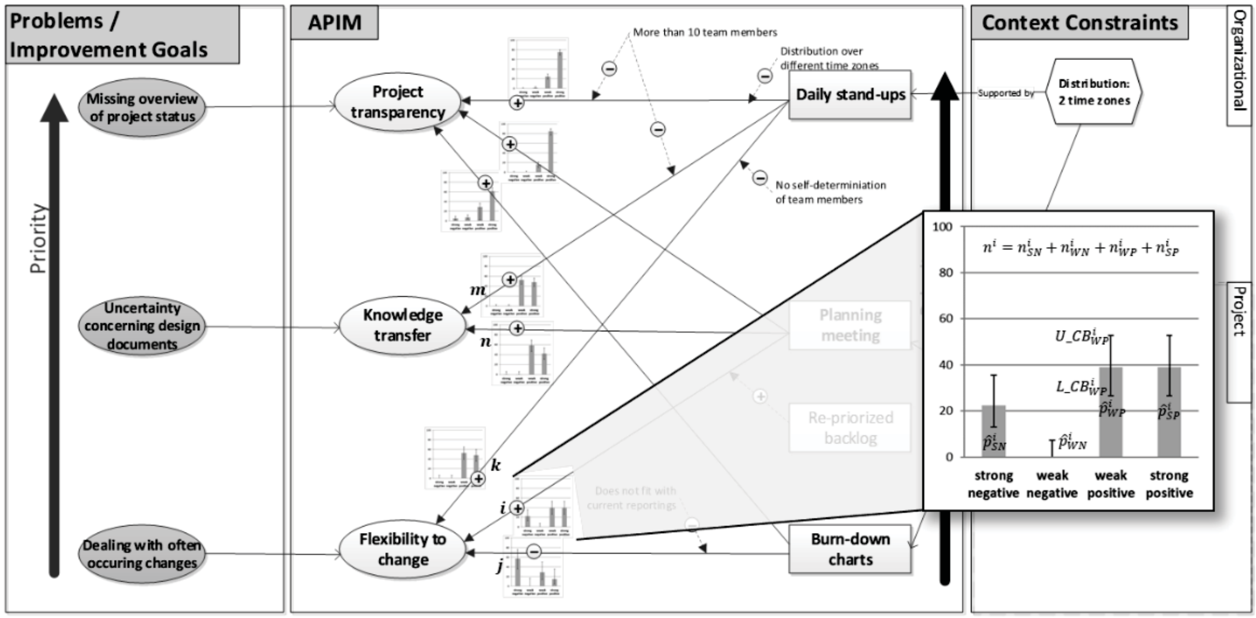


Figure 24: Example outcome within Step 2 - Activity 1 of the *Simulation of Process Improvements*

Now, the simulation results of all impacts i on the given process characteristics are equally summed for each simulation run and category. With another conversion from the absolute summed numbers to hit probabilities, it is possible to calculate the mean (μ) and the standard deviation. In our case, all the activity parts described above are performed using the statistical tool R, as it provides the possibility to sample directly from multinomial distributions.

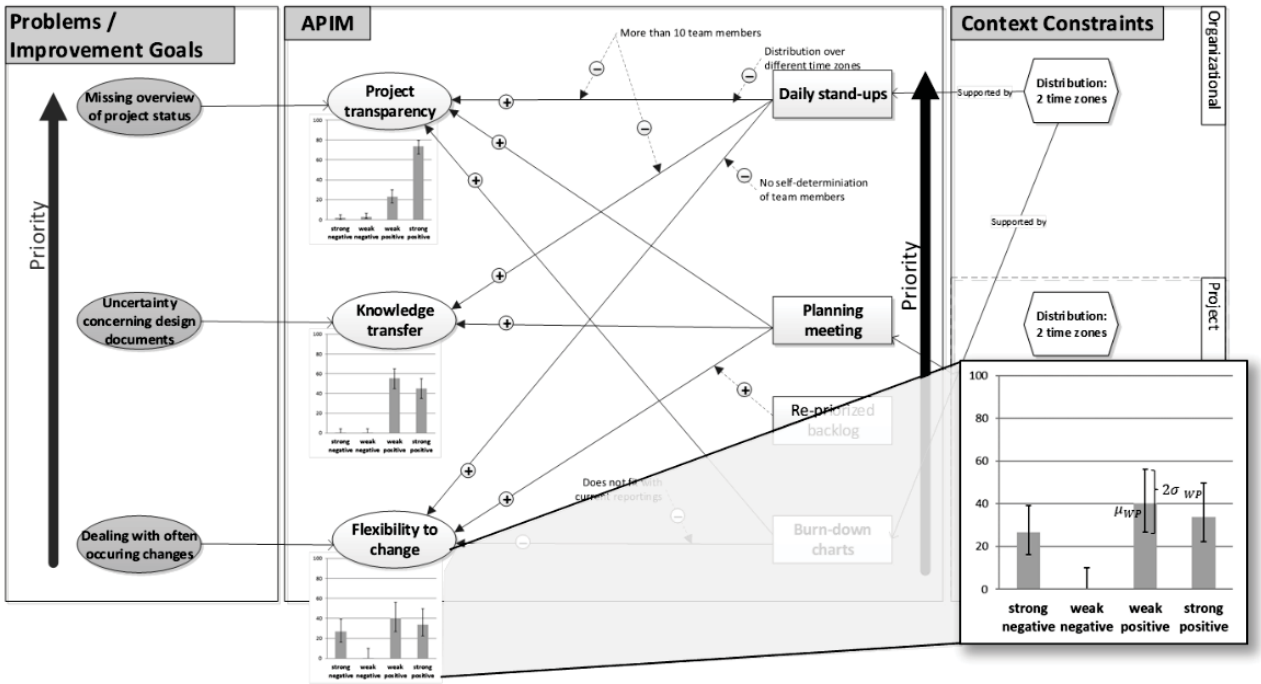


Figure 25: Example outcome after Step 2 - Activity 1 of the *Simulation of Process Improvements*

Activity 2.2: Similar to the visualization activity in Step 1, the activity "Visualize impacts for each improvement goal" uses the processed (aggregated) data and visualizes them such that the user gets a better understanding and to facilitate decision-making. In contrast to Figure 23, this visualization contains the mean (μ) as well as the standard deviation (σ) for every category. Especially for the standard deviation (Figure 25, black bar), it is possible to illustrate the single deviation around the mean or two times the standard deviation (2σ) (Chebyshev, 1867).

Activity 2.3: The activity "Derive worst- and best-case scenario" is necessary to improve decision-making at the end. We experienced that deriving a worst-case and a best-case scenario for every impact helps a lot with making decisions because it reflects the statistical uncertainty the decision maker faces when dealing with hit probability data. Summarizing these two cases briefly, in the best case scenario the two positive categories (strong positive and weak positive) should contain the highest possible value (close to the upper confidence bound, in the following denoted by U_CB_i) and the two negative ones (weak negative and strong negative) should have the lowest possible value (close to the lower confidence inbound, in the following denoted by L_CB_i). In the worst-case scenario, this is exactly the other way around.

This idea is limited by mathematical constraints, such as that all four categories must exactly sum up to 100%. Furthermore, the values for each category need to be within their corresponding confidence interval so that we can ensure that the values are reasonable estimates.

In the following, we will focus on finding the worst-case scenario. The best- case scenario is calculated analogously. The approach is formulated as a mathematical optimization problem where we set the objective function as

$$f(SN, WN, WP, SP\,) = \,-1.5\,SN - 0.5\,WN + 0.5\,WP + 1.5\,SP.$$

The four categories, respectively their probabilities (SN, WN, WP, SP) are the variables of the optimization problem. The coefficients ($-1.5, -0.5, +0.5, +1.5$) of the functions are due to the following two reasons:

- First, it was necessary to represent the directions positive (“+”) and negative (“-”) by the sign change.
- Second, an equal distance between the variables was necessary in order to establish an interval scale (Kirch, 2008) (Stevens, 1946).

The optimization problem is now stated as

$$minimize_{(SN,WN,WP,SP)}\; f(SN, WN, WP, SP)$$

such that

$$\left.\begin{array}{l} x \;\leq\; U\_CB_x \\ x \;\geq\; L\_CB_x \end{array}\right\} \;\forall x \in \;\{SN, WN, WP, SP\}$$

and

$$SN + WN + WP + SP = 1$$

This classical linear optimization problem can be solved using Danzig's simplex algorithm or method (Dantzig & Thapa, 1997). This algorithm can result in one of the following cases (Vanderbei, 2008) (Nering & Tucker, 1993):

1. There are infinitely many optimal tuples (SN, WN, WP, SP).

2. Exact one tuple (SN, WN, WP, SP) is optimal.

3. There exists no optimal solution.

The outcome optimum solution(s) for the minimization $minimize_{(SN,WN,WP,SP)}\ f(SN, WN, WP, SP)$ is our worst case. To find the best case, the objective function $f(SN, WN, WP, SP)$ has to be maximized instead of minimized. The constraints are the same. One possible tool that can be used for the optimization of best and worst cases is R. In these example cases, one single optimal solution was always found.

Activity 2.4: In the activity "Calculate agile practice contribution", the idea is to use the worst, normal (represented by the collected impact data), and best case of the different impact connections to derive how the practices contribute to the aggregated process characteristic (first and second activity). Therefore, we are using the same coefficients as in Activity 2.3:

$$-1.5\ SN - 0.5\ WN + 0.5\ WP + 1.5\ SP$$

Having the hit probability of the worst, most likely, and best cases, we can use these as variables for the four categories such that every impact (of an agile practice on a process characteristic) is represented by three values (one per case). Even if this calculation could have been done directly after

Step 1 as the first activity, as it is independent of the simulation, it becomes more interesting when we correlate these values with the overall impact on a process characteristic that was simulated before.

Activity 2.5: Similar to the previous visualization activities, the activity "Visualize agile practice contribution" presents the above-mentioned contribution of the agile practices and its relation to the overall impact. Figure 26 shows the single practices including their worst and best case together with the respective characteristic: Every horizontal row shows the contribution of a single practice to the characteristics (compared to the other practices influencing this). The thick bars represent the most likely contribution (Figure 26). Other than that, the range represented by the black line (Figure 26) gives the uncertainty derived from the worst and best cases; the shorter the black bar, the more reliable the most likely value is as a contribution of the single practice. Furthermore, having the means for each category of the aggregated impact on one process characteristic, these probabilities can be used as variables in the above-mentioned function. Thus, the dashed bars (Figure 26) crossing all contributing practices represent this value and serve as a reference for comparison. For the sake of understandability, we projected the values in this visualization on a numerical scale but re-translated the coefficients used in the simplex algorithm into the categories, e.g., weak negative as -1.5, on an ordinal scale.

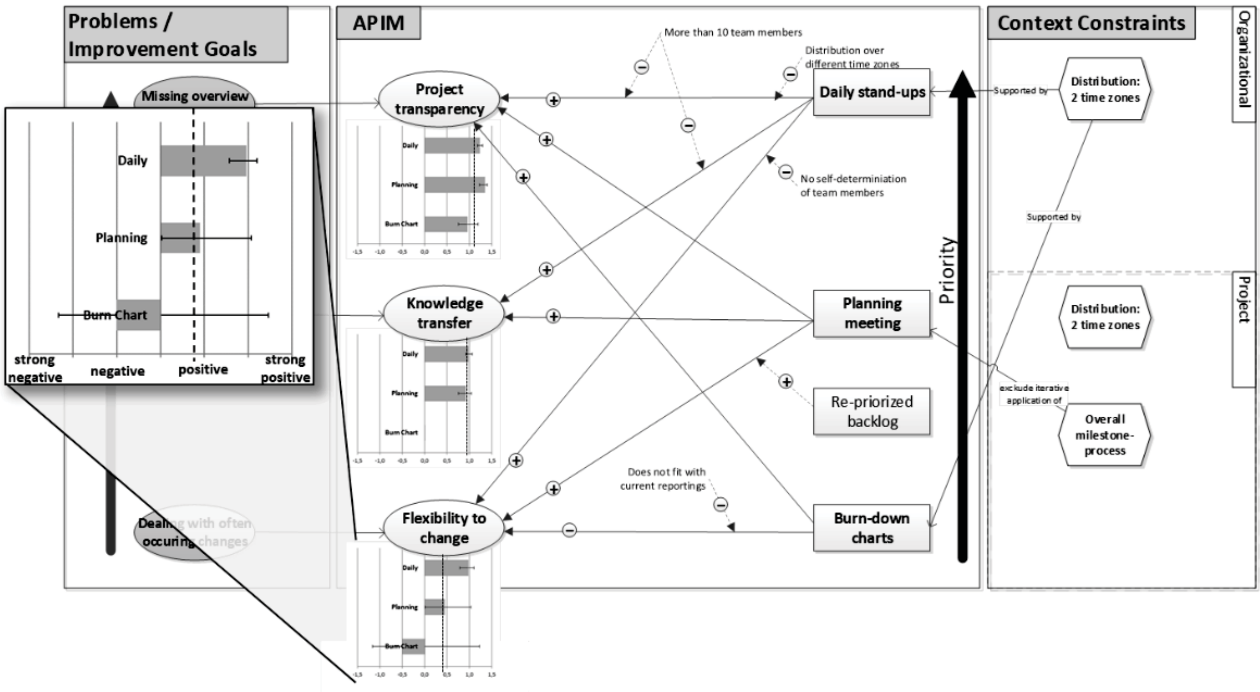


Figure 26: Example outcome after Step 2 - Activity 5 of the *Simulation of Process Improvements*

4.4.4 Step 3: Preparing Customer Quantifications for Simulation

Besides the two steps that work with the existing evidence data from the Agile Practice Repository and the APIM (cf. Section 4.2.2), the idea of Step 3 is to prepare the customer's own individual data and qualifications for the simulation. Thus, we collect the necessary data and transfer it to a simulation model together with the APIM. This works analog to CoBRA® (Kläs, 2011). It can be performed by one or several experts.

Table 14 summarizes the most important elements of Step 3. We provide a detailed description of each activity in the following comprising the "Simulating customer quantifications" steps.

Table 14: *Simulation of Process Improvements*: Step 3 – Simulating Customer Quantifications

Step 3: Simulating Customer Quantifications	
Objective	The objective of this step is to provide the possibility for an individual quantification of the impacts that can be used for a simulation to identify aggregated impacts on different improvement goals.
Inputs	• Tailored APIM • Evidence data on impacts (optional)
Activities	1. Check existing empirical impact evidences (from Steps 1 and 2) 2. Identify and select relevant impacts 3. Quantify selected relevant impacts 4. Build simulation model
Tools	• Tools for defining measures and quantifying the impacts on these measures. An example that can be easily used or adapted for our case would be the CoBRA® tool CoBRiX.
Outputs	• Simulation model: Combination of the tailored APIM with company-specific quantified impacts

Activity 3.1: The first activity, "Check existing empirical impact evidences (from Steps 1 and 2)", is the connection between the Agile Practice Impact Model data and its usage in the two previous steps with this second customer-individual simulation. Since in the following the company experts need to perform different activities based on their experience, it is very helpful to get to know this data and the results of working with this data. The checking of this data, as well as that of more specific data for individual domains, if available, should support the subsequent selections and quantifications done by the experts.

Activity 3.2: Based on this input, in the activity "Identify and select relevant impacts" the experts decide which agile practices do have an impact on different process characteristics. This identification is independent of the existing data. New impacts can also be identified or selected and existing ones can be ignored. In this activity, the focus is on selection, which forms the bridge to the next activity.

Activity 3.3: In the activity "Quantify selected relevant impacts", the identified impacts need to be quantified because this is necessary for the simulation. Therefore, at the end of this activity, there needs to be a quantification in a way that is (transferable to) a probability distribution. In CoBRA®, this is done using triangular distribution, which captures the experts' uncertainty with minimum, most likely, and maximum. Independent of the distribution (e.g., triangular distribution (cf. Figure 27), a scale is necessary. The scale definition needs to be transferable to a percentage scale and agreed among the company's experts. CoBRA®, for example, uses Likert scales for the elicitation of the experts' opinions that are easy to convert.

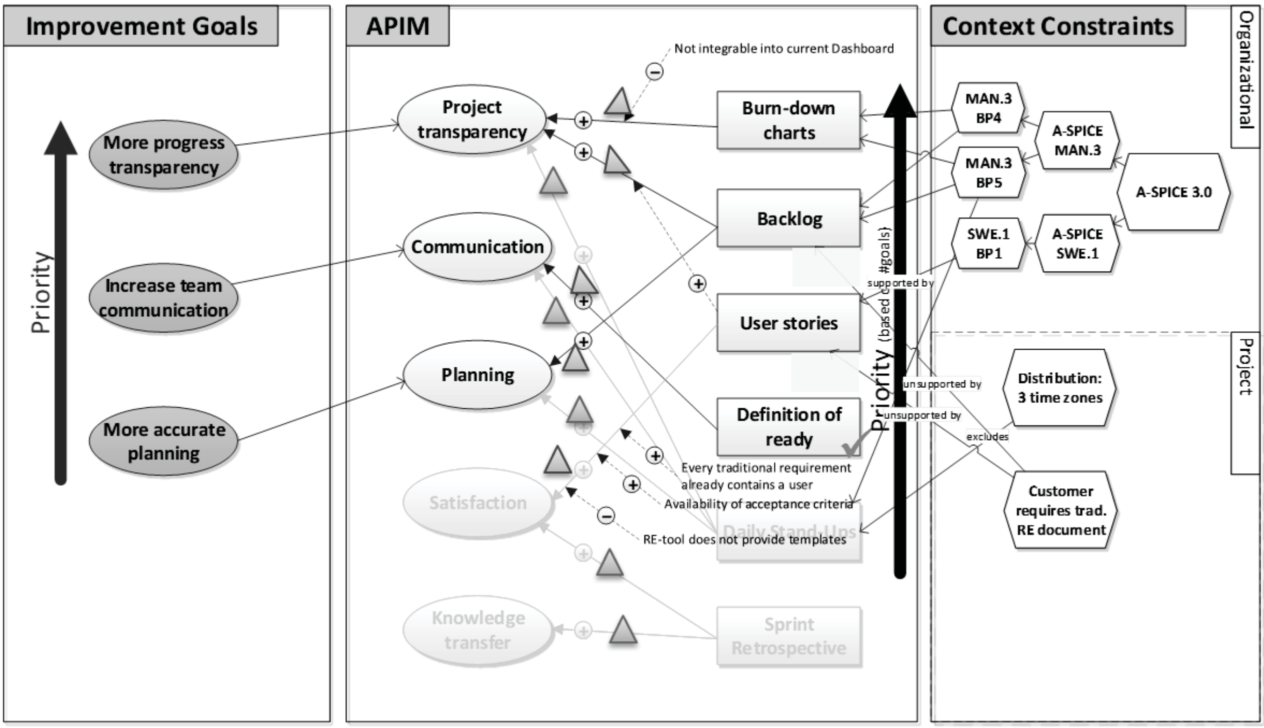


Figure 27: Example outcome after Step 3 - Activity 3 (triangles represent the maximum, most likely, and minimum probability density)

Activity 3.4: Finally, the activity "Build simulation model" brings the (tailored) APIM customer quantifications together and yields the simulation model. This means that the quantified impacts of all the participating experts are integrated into the respective impacts of the APIM (cf. Figure 27). This combination builds the simulation model for the customer-individual simulation.

4.4.5 Step 4: Simulating Customer Quantifications

As mentioned above, the idea of this step is the simulation of the previously created model as well as the analysis of its results. Similar to Step 3, most parts of this step work according to CoBRA® (Trendowicz, 2013).

Table 15 summarizes the most important elements of this step. We will provide a detailed description of each activity comprising the “Analyzing Customer Quantifications” step below.

Table 15: *Simulation of Process Improvements*: Step 4 – Analyzing Customer Quantifications

Step 4: Analyzing Customer Quantifications	
Objective	The objective is to run the simulation based on a Monte-Carlo simulation such that a subsequent sensitivity analysis can be used to decide on possible contextual changes or (re-) prioritization of the suggested agile practices.
Inputs	• Simulation model
Activities	1. Run simulation 2. Perform sensitivity analysis 3. <optional> Analyze contextual changes 4. Derive final transition backlog
Tools	• Tools that integrate a Monte-Carlo simulation and can read the simulation model as input. An example is the CoBRiX tool, used for the cost estimation simulation CoBRA, which works in a similar way.
Outputs	• Simulation results • Possible contextual changes • List of prioritized agile practices for process improvement → Transition backlog

Activity 4.1: The core of this step is the activity “Run simulation” because it is the activity in which the simulation model created in the previous step is executed. For running the simulation, we use Monte-Carlo simulation (Doubilet, et al., 1985), which was already used previously in the non-customer-specific simulation steps. The detailed sample in the simulation depends on the number of participants in the previous step building the model. If more than one expert participated in the previous step, a simulation run contains several samplings for each process characteristic, similar to CoBRA (Trendowicz, 2013). In cases where only one expert participated, only the latter sample is necessary.

Activity 4.2: The activity “Perform sensitivity analysis” produces the contribution of the individual agile practices to one characteristic. The only difference to Step 2 - Activity 5 is that it only presents one value instead of a range of values. A sensitivity analysis is the study of how the uncertainty in the output of a mathematical model or system, in our case the simulation model, can be apportioned to different sources of uncertainty in its inputs (Saltelli, 2002) (Saltelli, et al., 2008). Of its different purposes

(Pannell, 1997), in our case the sensitivity analysis should increase the understanding of the relationships between input and output variables of the model, identify important connections, as well as enhance communication with decision makers, e.g., by making recommendations more credible. An example outcome of a sensitivity analysis is depicted in Figure 28.

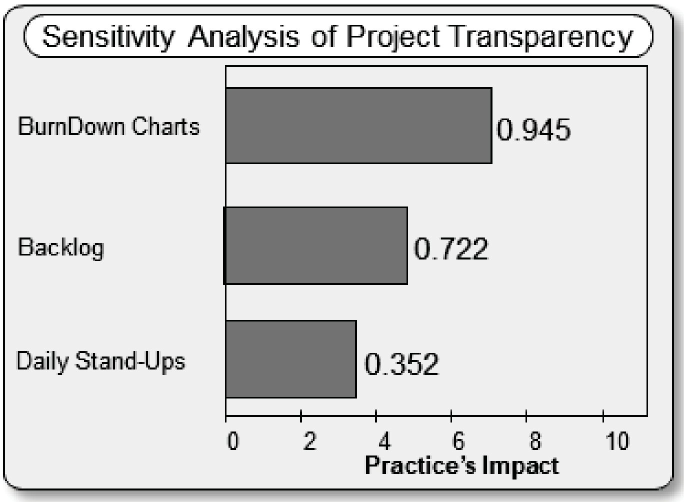


Figure 28: Example outcome after Step 4 – Activity 2

Activity 4.3: The aim of the activity "Analyze contextual changes" is to check how changing the context might influence the results of the selection of appropriate agile practices. Since the simulation could also be executed easily with agile practices originally excluded by the *Agile Potential Analysis*, it is easy to run the simulation with the originally excluded agile practices. Seeing how these practices behave and examining the results of the previous sensitivity analysis may make it worthwhile to think about changing the context such that agile practices with high impact are used. This is an optional activity because it needs to be performed iteratively with the previous activity if the full power is to be achieved. Furthermore, it also requires the experts to quantify all previously excluded agile practices that should be included in the analysis of contextual changes as well as the linked simulation.

Activity 4.4: Finally, the idea of the activity "Derive final transition backlog" is to (re-)prioritize the transition backlog, respectively its backlog items (the agile practices resulting from the *Agile Potential Analysis*). The quantifications in Steps 1 to 3 and the activities in Step 4 should support the decision-making regarding the suggestions of the *Agile Potential Analysis*. Furthermore, especially the previous activity with possible context changes should help to question the previous transition backlog and might result in a new one, with different elements or re-prioritization as a result of having and knowing quantitative data.

4.4.6 Tool Usage

The tool parts of the simulation need to be separated into the first part of the simulation (Steps 1 and 2), which can be performed with every execution of the *Agile Potential Analysis*, and the second part (Steps 3 and 4), which is optional for customer-specific impacts and simulation and even more effort-intensive.

The first tool we used was “R”, which is used for statistical computation. By creating a small script, it was possible to read the CSV with all the collected impact data, calculate the simultaneous confidence intervals using the Goodman method, and finally write this calculated statistical data into another CVS file. In general, this activity could also be tool-supported by other tools for statistical calculations, such as SPSS or others.

The first simulation was performed with a combination of “R” and Excel. R (Venables, et al., 2018) or other equivalent statistic tools are necessary for creating the worst and best cases using the simplex algorithm. In contrast, Excel can be used for the creation of random numbers and simulation aspects. We selected these tools because they are powerful enough for the calculations and can provide random sampling of values for the Monte-Carlo simulation. Last but not least, it is easy to create diagrams from the different data sets. For both of these tool-usage parts, it would also be possible to create a specific tool that only performs the above-mentioned activities. Since we are not quite sure how often this simulation add-on for the *Agile Potential Analysis* will really be used or requested, we decided to work with the above-mentioned “tool chain” and not implement any specific tooling. Nevertheless, it would be possible to integrate parts of these steps or activities into the Excel tooling of the *Agile Potential Analysis* (cf. Section 4.3.7).

Finding tool support for the customer-individual simulation, which is the optional and second part of the overall simulation, was much easier. Since this part is built upon new customer-individual data and our APIM, which is similar to the CoBRA® model (Trendowicz, 2013), we were able to reuse their CoBRiX tool[11] (Kläs, 2011). Due to the fact that the APIM already exists before the simulation, the steps until the quantification of the factors of CoBriX can be skipped. The quantification of the factors or variables is followed by the collection of multiplier data, where the experts can specify their impact data. The following steps – validating the multiplier data, building the model, and validating the model – can then be performed in a similar way as in the standard CoBRiX tool (Kläs, 2011).

4.4.7 Summary

Summarizing the *Simulation of Process Improvements*, we see the need for and benefits of both parts, the generic simulation using the existing data of the Agile Practice Repository and more specifically the APIM as well as the customer-specific quantification. Figure 29 visualizes the coherences between the different steps as well as their activities, similarly to Figure 21 with the inputs and outputs of the activities.

[11] https://cobra.fraunhofer.de/cobrix/index.html

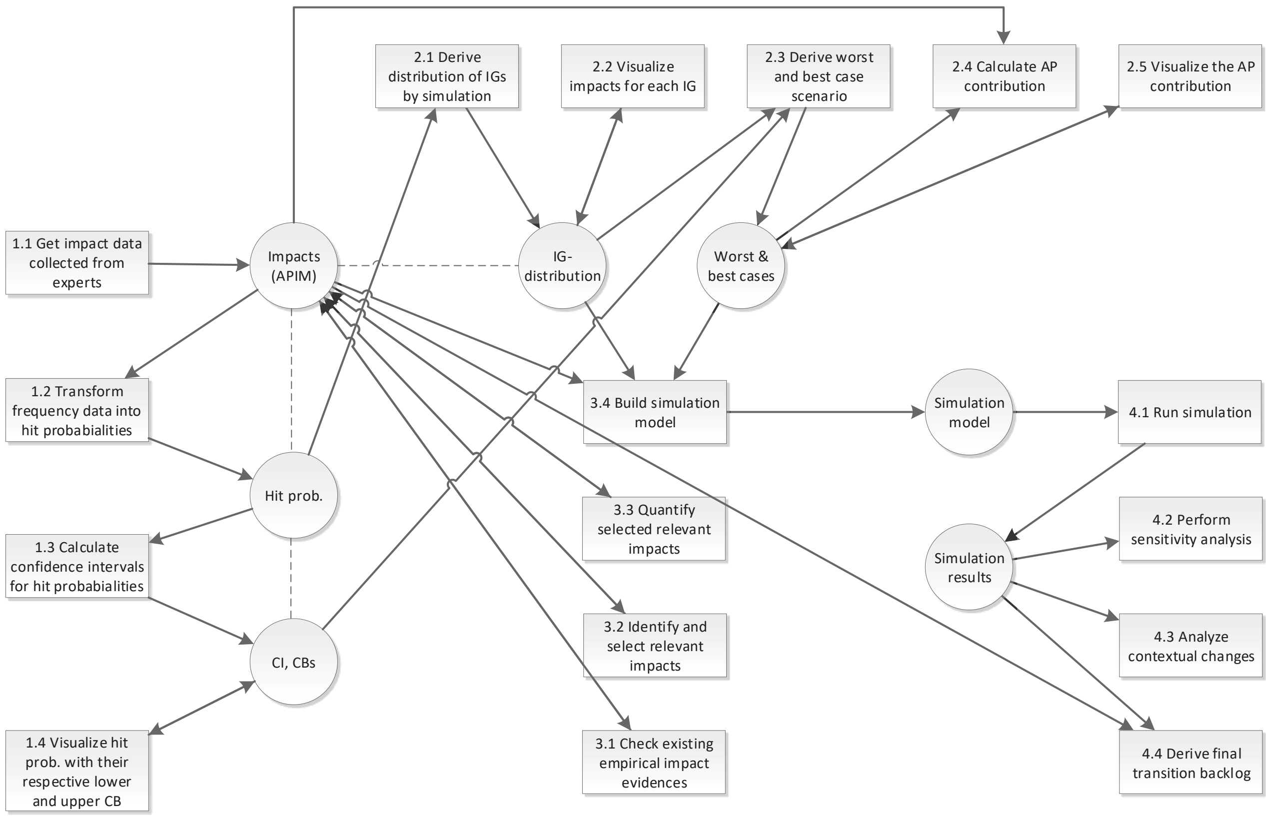


Figure 29: Detailed illustration of the sub-steps of the *Simulation of Process Improvements* including inputs and outputs (Step 1 on the left, Step 2 at the top, Step 3 in the middle; Step 4 on the right)

The incoming and outgoing edges in Figure 29 show the impacts as well as their hit probabilities as the major artifacts. Especially the impacts in combination with their contribution form the connection between the two simulation parts. Figure 29 shows that Step 1 is more of a preparation step, ensuring that the simulation in Step 2 is performed smoothly. A similar picture is shown with Steps 3 and 4, which only consider customer-individual impacts. Especially Steps 3 and 4 are the ones we will be using with the Experience Factory approach (Basili, et al., 2008), as these company-specific experiences are integrated into our existing experience base (the Agile Practice Repository with the APIM) in an anonymized way. Thus, this improves the first part of the simulation as well as the *Agile Potential Analysis* using this data.

The *Simulation of Process Improvements* in general adds some additional effort to the effort for performing the *Agile Potential Analysis*. Nevertheless, we need to distinguish between the two parts of the simulation (Steps 1 & 2 and Steps 3 & 4) as well as between the roles for which effort arises. In general, the first part is less effort-intensive because the data exists already and can be used for calculation and analysis. In contrast, the second part also includes the expert quantification with its initial elicitation (Step 3). This is the only part of the complete simulation where effort for the customer experts is necessary, which might be much effort due to the fact that the reliability of the simulation results increased with the number of experts. Even though the last section explained all the existing tool support (Section 4.4.6), some manual effort needs to be performed by an expert in the *Agile Potential Analysis* or at least in the model(s) and data used in the simulation, which are the results of the analysis. In summary, the first part of the simulation requires little effort and is therefore a nice add-on, whereas the second part brings a more individual perspective that is more effort-intensive. Nonetheless, the usage of the existing CoBRA® approach makes it much easier to create and run the simulation . Furthermore, the experiences of Fraunhofer IESE with the simulation approach of CoBRA® have shown that even expending some effort is worthwhile.

4.5 Limitations & Requirements Coverage

Before we discuss the requirements defined at the beginning of this thesis, it is important to mention that the overall method as well as its two parts support the agile team and project levels of a complete agile transition, but not complete business units or a complete enterprise. Even if our repository contains some scaling practices like Scrum-of-Scrums or Communities-of-Practices, they are more focused on (very) large projects and not on organizational units or whole organizations.

Since the remaining limitations are not given by any external aspects, such as the size of the company or its agile transition, we will now discuss the major limitations according to the requirements presented in Chapter 2.3. Table 16 gives an overview of all the strengths (indicated by "+") and weaknesses (indicated by "-") of the Agile Process Improvement Approach (cf. Chapter 4).

Table 16: Assessing the approach with respect to the requirements in Chapter 2.3

Requirements	**Assessment**	**Description**
REQ: goal-/problem-related SPI		
Relationship to org. goals	+	The *Agile Potential Analysis* works on the APIM, which connects the suggested agile practices with process characteristics that are related with the organizational goals.
Stakeholder involvement	o	For both parts of the method, it is very helpful to involve many different stakeholders in the performance such that their viewpoints are covered when considering the context. Nevertheless, it is not mandatory to do this and works only with a lower number of stakeholders.
REQ: evidence-based SPI		
Transparent decision-making	+	Having a well-described process for the *Agile Potential Analysis* and the *Simulation of Process Improvements* should help to make the suggested results and the way to arrive at them transparent. Only some parts, such as the algorithm for the prioritization, are not given to customers. However, the transparency depends on the time people spend understanding or going through the explanatory material.
Evidence repository	+	The approach provides and uses an extensive repository of agile practices as well as the APIM covering the mapping of the practices to the impacted characteristics.
Quantitative consideration	o	Besides the usage of the experts' data, which is not that obvious or transparent in the analysis part, mathematical calculations and simulations in the other part provide a significant quantitative contribution.
REQ: continuous SPI		
Step-by-step evolution	+	The prioritized transition backlog only contains single agile practices and no complete suggested agile methods, such as Scrum. Even though the top practices are all Scrum practices, they are given in an order for stepwise implementation.

Reversibility of changes	+	Easy reversibility is ensured by introducing only single agile practices; furthermore, the repository contains Undo steps for the different practices.
Suggesting improvement actions	+	The agile practices with possible adaptations that are the backlog items of the transition backlog serve as improvement suggestions.
Context REQ		
Consideration of reg. req.	+	The *Agile Potential Analysis* covers regulatory requirements as part of the steps dealing with the context (Steps 2 and 3). Furthermore, the analysis method already includes some specific regulations, such as Automotive SPICE.
Consideration of context issues	+	The analysis method considers the context in two dedicated steps that highly influence the selection and prioritization.
Maturity	+	With a large repository including the experts' evidences, which also work as a marketing teaser, the overall approach was designed for easy application in practice. Only the optional last part of the simulation might be too effort-intensive for potential customers.
Rigor	+	Due to its concrete steps with inputs, detailed activities, and outputs, which are often enriched with examples, the Agile Process Improvement Approach is very concrete.

The analysis above in Table 16 mainly reveals two basic limitations of the current approach:

- **Stakeholder involvement:** Currently, it is very helpful to involve different stakeholders during the *Agile Potential Analysis*, especially in Steps 2 and 3 where the context is considered. This is the case because different viewpoints help to identify as much contextual information as possible, which is not the case if there are fewer stakeholders with a limited view. Since the approach does not have any mechanism for specific stakeholder integration, future versions of the approach should consider this aspect. Since context elicitation is currently performed with the help of a list of important aspects to consider, one possibility would be to cluster these aspects according to specific stakeholders.

- **Quantitative consideration**: Ins the field of software and system development processes, considering data is often a tough job, especially when it comes to the aspect of measuring the effect of process improvements. We could only collect and use subjectively perceived data from experts instead of measuring something objective. Nonetheless, the method would also work with measured objective data. In general, there is a lack of such data in all state-of-the-art approaches that integrate some quantitative data. The SAAF (Esfahani, 2015), for example, only provides quantitative data from one single case study conducted during the development of the approach.

Comparing the approach with the state-of-the-art approaches assessed according to the same requirements in Chapter 3.4, the approach of this thesis shows better coverage of the requirements. It especially stands out with regard to the requirements of evidence-based SPI, such as the evidence repository, as well as the context requirements considering different contextual aspects.

4.6 Summary

In this chapter, the Agile Process Improvement Approach for better decision support for an evolutionary process improvement initiative using agile practices was introduced:

First, the underlying repository and model used in the approach were presented together with the necessary data containing 70 practices with a detailed description. The repository was created during the course of several projects with different agile experts. The created model, which represents a cause-effect relationship, contains more than 3800 opinions collected from agile experts all over the world. Second, the first part of the overall approach, the *Agile Potential Analysis*, was described and partially illustrated with examples. It is an analysis method consisting of four steps that uses current problems as well as the specific organizational and project context to identify, prioritize, and adapt appropriate agile practices. Based on this output, a step-by-step transition towards the right degree of agility for the specific context is possible. Third, the *Simulation of Process Improvements* was introduced, which extends the *Agile Potential Analysis* with quantitative data such that the final decision regarding improvements by using agile practices is improved. To reach this goal, the existing data collected by experts is processed using different mathematical and statistical methods as well as a simulation to provide reliable visualizations to the users. This part of the overall method also offers the possibility to create and use one's own data for the simulation, which provides more individual results but is also much more effort-intensive.

Compared to the related state-of-the-art approaches, the unique selling points of our approach are:

(1) the evidence repository with its quantitative consideration (evidence-based SPI requirements);

(2) the consideration of context issues and regulatory requirements (context requirements);

(3) its maturity;

(4) its rigor.

Overall, this approach has achieved the integration of most of the advantages of the related approaches. In the assessment of the approach, ten requirements were found to be addressed well, whereas the two remaining ones are partially addressed. With respect to the state of the practice, we see the benefit in the creation of appropriate individual degrees of agility for a dedicated context.

5 Empirical Validation

"The best evaluation I can make of a player is to look in his eyes and see how scared they are."
Michael Jordan

This chapter includes the empirical validation of the ACAPI approach in real practitioners' setups. Beginning with a detailed validation methodology (Section 5.1), the two subsequent sections present the validations of the *Agile Potential Analysis* (Section 5.2) and the *Simulation of Process Improvements* (Section 5.3). Finally, the results of these studies will be compared and discussed, and all empirical validations will be summarized (Section 5.4).

5.1 Validation Methodology

The validation of the ACAPI approach was performed and packaged based on the case study guidelines of Runeson et al. (Runeson & Höst, 2008) (Runeson, et al., 2012) and the empirical guidelines of Jedlitschka and Pfahl (Jedlitschka & Pfahl, 2005). Figure 30 summarizes the validation methodology in terms of goals, hypotheses, and performed studies.

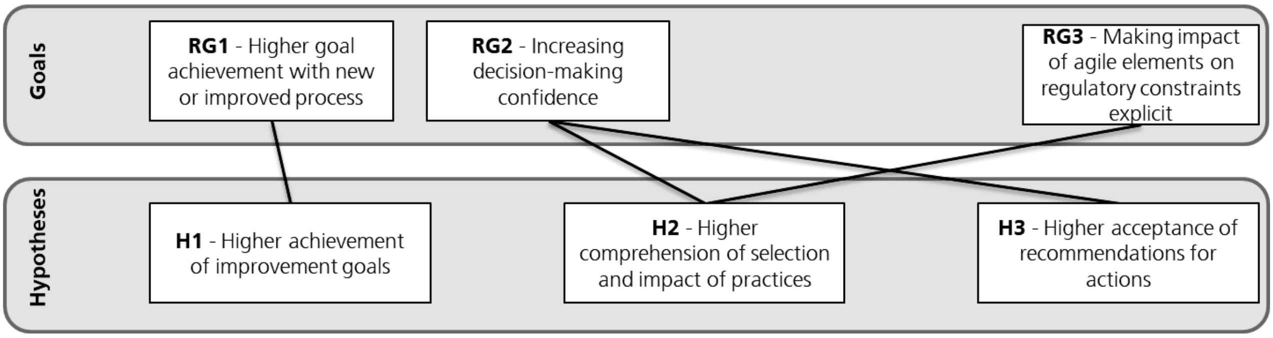


Figure 30: Mapping of goals and hypotheses

5.1.1 Goals and Hypotheses

The following evaluation hypotheses were established based on the three research goals RG1 to 3 (cf. Section 1.5.1) defined at the beginning of this work (cf. Figure 30):

H1 – Higher achievement of improvement goals: When introducing a new process or improving it, companies expect to achieve some improvement goals. Examples of improvement goals include faster time to market, increasing quality, or more transparency. The improvement goals vary among different companies; moreover, they can vary across different project teams within a company. The overall approach aims at increasing the achievement of improvement goals independent of the context (RG1).

H2 – Higher comprehension of selection and impact of practices: This hypothesis is driven by both remaining RGs, increasing decision-making confidence (RG2) and making the contribution of agile elements explicit (RG3). Thus, the second aim of the approach is to increase comprehension of the selection of improvement actions, in our case the agile practices, as well as their impacts. According to (Guzman, et al., 2017), comprehension means the combination of understandability, transparency, and reliability. Thus, the aim is to increase these different aspects.

H3 – Higher acceptance of recommendations for actions: This hypothesis is derived from RG2, dealing with decision-making confidence. It shares the same research goal and is thus connected with hypothesis H2.

The aim of increasing the acceptance of the recommendations resulting from the approach presented in this thesis is refined by job relevancy, results demonstrability, output quality, perceived ease of use, and perceived usefulness (Guzman, et al., 2017).

5.1.2 Strategy

As presented in Chapter 4 the overall method developed in this thesis is a combination of two parts, the *Simulation of Process Improvements* and the *Simulation of Process Improvements*. They were evaluated slightly differently in terms of the detailed research design and execution. Both cover all three goals and the related hypotheses of the previous section (see research designs in Section 5.2.1 and 5.3.1), as can be seen in Figure 31.

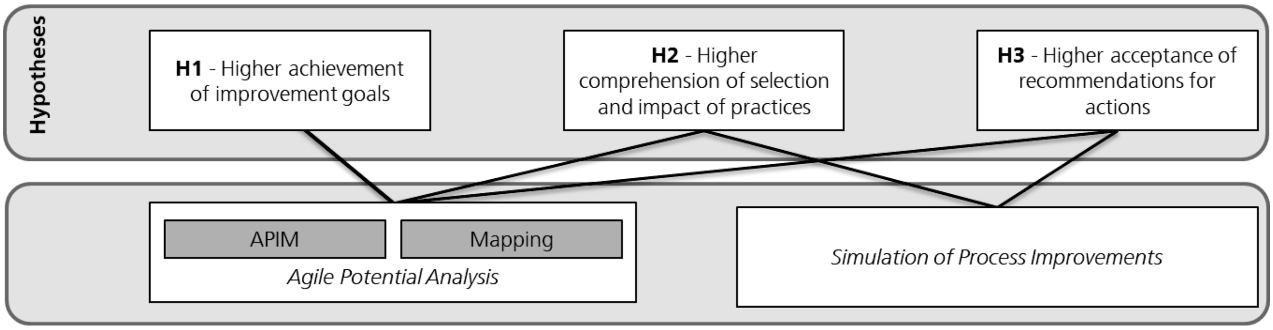


Figure 31: Mapping of hypotheses and studies / solution parts

To evaluate the *Agile Potential Analysis*, case studies were selected as the evaluation method and strategy based on the following considerations:

- There is no comparable state-of-the-art approach (cf. Chapter 3). Since no comparison with other approaches is possible, controlled experiments were excluded.

- A minimum background in software engineering, agile experience, and the application domain is needed to apply the ACAPI approach. With regard to software engineering, knowing the development lifecycle with its phases and common engineering activities as well as having an idea of agile practices is the prerequisite. Furthermore, selecting and introducing agile practices requires a lot of experience in the respective application domain, including common standards and their influences.

For this reason, the decision was made to design and perform case studies (Runeson & Höst, 2008). Case studies are especially useful for real industrial set-ups that do not allow complete control of the environment (Diebold, et al., 2016). Thus, we evaluated the *Agile Potential Analysis* by performing six case studies in six different companies using common evaluation guidelines.

To evaluate the *Simulation of Process Improvements*, a walkthrough was designed and performed (Wharton, et al., 1994) with experts who had participated in the case studies of the *Agile Potential Analysis*. Walkthroughs are especially useful when it is hard or impossible to perform the given method, e.g., because it might be too time-consuming, and when it is easy to create a scenario using the method.

These decisions resulted in the overall validation procedure presented in Figure 32 below. The detailed procedure for the two parts of the ACAPI approach will be described separately: in Section 5.2 for the *Agile Potential Analysis* and in Section 5.3 for the *Simulation of Process Improvements*.

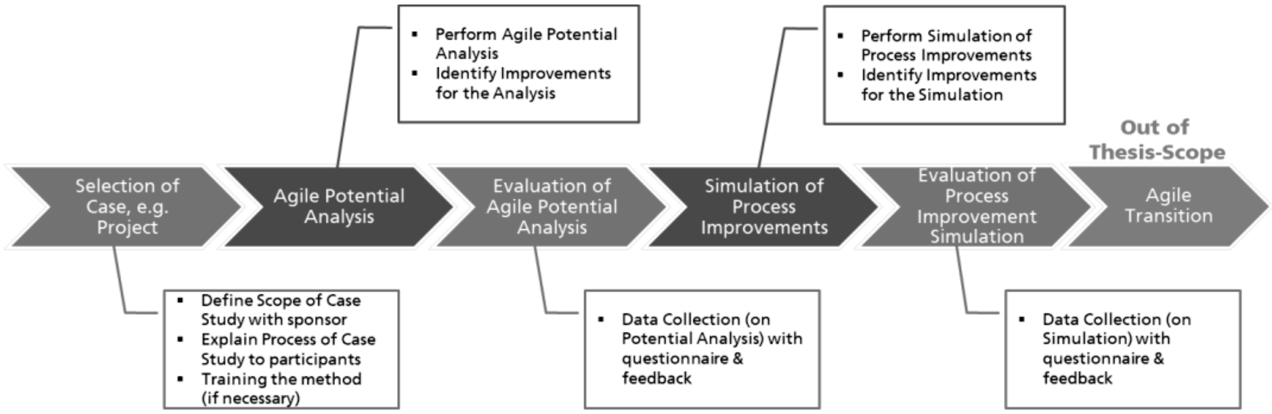


Figure 32: Overview of the validation procedure

5.2 Validation of Agile Potential Analysis

This section presents the concrete validation for the *Agile Potential Analysis*. It covers all aspects from the research design to the data analysis and interpretation.

5.2.1 Research Design

In the evaluation of the *Analysis Potential Analysis*, our focus was on following the procedure presented in Figure 32. Prior to the start, the company sponsor[12] as well as the participants were informed about the procedure.

Case selection: The scope of the case study was defined together with the company sponsor, e.g., the project manager. The participants were selected according to their availability and willingness to participate as well as according to their knowledge on software engineering.

Agile Potential Analysis: After a short explanation of the case study procedure, a first baseline was collected on current problems and goal

[12] person from the (industrial) company paying for or responsible for conducting the analysis

support. Using this baseline, the *Agile Potential Analysis* was performed on the customer's case, as described in Section 4.3.

Feedback: After the complete execution of the analysis, data collection and feedback took place. Feedback was elicited by means of a questionnaire (cf. Section 5.2.5) presented to the participants either at the end of the result presentation of the analysis or afterwards.

The study procedure will be refined in more detail in the following subsections.

5.2.2 Population and Sample

The target population consisted of roles from a crosscut of a common software development project. Since roles largely vary across companies, we mapped existing roles in each company to the roles defined in the *Agile Potential* Analysis (APA), i.e., APA Executer, Process Owner, and Process User[13].

The sample included participants from six different companies. With the exception of Bosch from the Automotive domain, all were from different information system domains. Table 17 provides an overview of the companies and the participants.

Table 17: Population and sample of *Agile Potential Analysis* case studies

Company	Context	Time	# Participants		
			APA Executer	**Process Owner**	**Process User**
VSF Experts	SME	Q2-3 / 2017	1	0	1
Capgemini	Enterprise	Q4 / 2017 (Beg.)	1	1	7
TQsoft	SME	Q1 / 2018 (Beg.)	0	1	4
Kemweb	SME	Q1 / 2018 (Beg.)	1	1	5
Yatta	SME	Q1 / 2018 (Beg.)	0	1	1
Bosch	Enterprise	Q2 / 2018	0	1	0

[13] Person that uses and executes some of the activities specified in the process, such as a developer, tester, requirements engineer, etc.

The companies involved in the evaluation of the *Agile Potential Analysis* varied with respect to application domain, company size, company structure, customers, regulations, etc., as described below:

Case 1: **VSF Experts** are an SME that has three development teams with about seven people each. Development is distributed across several locations in two countries: Germany and Poland. Regarding their project structure, the participants mentioned that they are working in a multi-project environment such that normally each team is working on several projects in parallel.

They stated that they are using or adapting Scrum to their specific needs. Since they mentioned that the requirements and architecture is done in Germany and especially the source code is coded in Poland, the structure partially resembles a waterfall-like process. In addition, on their website it says that they also do some external coaching on agile development, meaning that at least some employees should have experience with that.

Case 2: The overall context of the **Capgemini** case study was the global IT of one of their customers, a logistic company. Capgemini staffed one their development teams, with 14 team members. This dedicated team formed the scope of this case study.

Even though this is in an information system development, the team is regulated to some extent: Their project manager mentioned the ISO25010 (International Organization for Standardization, 2011) on software product quality. Capgemini and their customer had relatively strict guidelines on a waterfall-based procedure for the project, including milestones etc. The overall project used a traditional process with some iterations, due to the waterfall guideline(s). Nevertheless, this particular team had already tried some agile practices. A few, mainly technical, practices such as continuous integration, automated test, automated builds, coding standards, or refactoring, were still in use at the time of our evaluation. Some team members mentioned that they also tried to use other agile practices, but selected them without any specific reasons and used them without any explanation.

Case 3: The detailed context of the **TQsoft** case study is given in (Diebold, et al., 2018). With the help of a so-called "Informer", they develop their own workflow software to support different business processes. Based on this standard product, the majority of their staff is working on customer projects related to their software. With their core team of six people, they are the smallest company among our studies. We found that no knowledge existed with regard to agile development, even though they were already using a few adapted agile practices without any explicit informed introduction.

Case 4: **Kemweb** was the largest SME in this validation, with a total of 35 to 40 employees. They are a media and web development company. With their core competencies being in the areas of (more complex) website development, they are structured according to their different disciplines, e.g., UX, backend, frontend, etc. Due to the dependencies between these and the need to wait for each other, most of their projects are conducted in a waterfall manner.

During initial discussions it turned out that they do not have a standard project because every project is different. This is also one reason why the project team, in its multi-project setup, varies for every project and is established new for every project. Furthermore, they often have requirements changes that should be motivation for agility. This was also one of the reasons why the CTO started some work in this direction, e.g., by getting training as a professional Scrum Master.

Case 5: **Yatta** with its 25 employees works in a very specific domain. As an information systems development company, they focus more or less exclusively on Eclipse-related software. They have several fixed development teams for the projects they are dealing with. In addition, a smaller part of their work is their consulting service, which also centers around

Eclipse. From their experience and knowledge in processes and agile development, we knew that they are applying some agile aspects. In their case, the teams are free to decide on how of these aspects they want to employ, meaning this ranges from only some elements to complete agile methods, mainly Scrum.

Case 6: Bosch employs several thousand employees and developers. As it is such a large enterprise, we were working with their business unit that is working in Automotive, an embedded systems domain where highly safety-critical systems are being developed, e.g., brake systems. This also means that regulations such as Automotive SPICE and ISO26262 for functional safety are important to be considered in their context.

The specific project that was considered in their case study included several teams (more than ten) distributed across three or more different countries. They built a new cross-cutting sub-organization for this project over the current business units in their automotive area. For this, they chose a matrix organization with a strong product focus. Already familiar with agile development to some extent because of global initiatives, they were already using some very technical agile practices such as continuous integration.

5.2.3 Execution

As shown in Table 18, all six case studies were executed between the third quarter of 2017 and the second quarter of 2018. Since the execution of the *Agile Potential Analysis* varied slightly between the different case studies, this section gives an overview of the general execution with some examples and variations. These variations stem either from the specific context, e.g., availability of the necessary people, or from lessons we had learned from earlier studies.

The main variation factor and possible confounding factor during the execution was the executer performing the analysis. The executers had different backgrounds and approaches for presenting the results and performing the elicitation. Table 18 shows all these factors for the different case studies.

Table 18: Differences in the execution of the *Agile Potential Analysis* case studies

Company	**Duration of APA**	**Executer**	**Result Pres.**	**Performance**
VSF Experts	~ 2 months	Company student	Model	Step 1: Internal interviews Steps 2 & 3: One workshop with IESE
Capgemini	~ 1.5 month	Student (on site)	Table	Steps 1 - 3: Four internal Interviews (with team lead, assistant TL, two developers)
TQsoft	~ 1 week	P. Diebold	Table	Steps 1 - 3: One workshop
Kemweb	~ 1 week	P. Diebold & CTO	Table	Steps 1 - 3: One workshop
Yatta	~ 1 week	P. Diebold	Table	Steps 1 - 3: One workshop
Bosch	~ 2 weeks	P. Diebold	Table	Steps 1 - 3: One interview (3 participants, by IESE) + slides

The first two case studies lasted longer (up to 2 months) than the other ones (up to 2 weeks) because they were conducted by or in cooperation with a student during his thesis. Because of the duration of the thesis and because the major effort was for the student, not for the company, they took more time and could go into more details, which is also shown by the fact that interviews took place instead of workshops. We also observed that the duration varied depending on the size of the organization/team/project involved during the *Agile Potential Analysis*. Furthermore, the duration also depended on the availability of the participants for the different steps, from the initial elicitation to the final presentation.

Regarding the representation and tooling aspect, we generally distinguish between a visual model representation (also used in Chapter 4.3, cf. Figure 15 - Figure 18) and a tabular representation (cf. Table 11). This representation aspect is strongly connected with the tooling aspect. With the model, we can work in modeling tools like Enterprise Architect (where we created an add-on) or Microsoft Visio, whereas for the tabular representation, simple spreadsheet tools like MS Excel can be used (cf. Chapter 4.3.7). Table 18 shows that we only used the model representation once and learned that it is getting too large and complex to keep it in that form (especially because we started on a whiteboard). For that reason, we switched to the other representation for the remaining case studies.

5.2.4 Procedure

The *Agile Potential Analysis* could be performed differently, especially regarding the elicitation and discussion of the necessary inputs (improvement goals and context). This could be done with any kind of elicitation technique. Nevertheless, we recommend (and used ourselves) either interview(s) or a workshop, or a combination of both. While we used interviews more for the pure elicitation, some of the detailed activities of the analysis steps could be discussed directly in a workshop, such as understanding of the context and directly linking it to the APIM model (e.g., in Steps 2 and 3). This was the reason why we used workshops in all the case studies we executed. Independent of the method used, the results of the different steps look quite similar and will be presented briefly below, exemplified by the Capgemini case study:

Step1 – Connecting Improvement Goals: Figure 33 illustrates the results of the first step: The identified current problems were prioritized and mapped to our process improvement characteristics. We used them to create the first list of prioritized agile practices just on the prioritized improvement goals.

1. Problems	2. Transfer to Improvement Goals (formulate positive)	3. Vote the Problems (Higher Number -> more important)	4. Process Characteristics: Customer collaboration	Productivity	Knowledge transfer	Transparency (of project status)	Customer value	time to market	Internal communication	Internal documentation	Foster agile culture	Throughput	Flexibility and ability to change	Team Spirit	employee motivation	Product quality	Team synergy	Development Cost
			3	0	2	1	0	0	0	0	0	0	4	0	0	5	0	0
Product quality could be higher	higher product quality	5														X		
Long time intervals of developing (sub-)releases	more flexibility	4											X					
Long feedback cycles	better customer collaboration	3	X															
Sometimes uncertainty concerning design documents	better knowledge transfer	2			X													
Sometimes missing overview of project status	better transparency of project status	1				X												

Figure 33: Capgemini case study - *Agile Potential Analysis* Step 1

Step 2 – Analyzing Practices with respect to the Context: In Step 2, we mainly excluded practices because they were either being used/had been tried already or did not fit the context. These practices were filtered out in the spreadsheet before Step 3 was performed. In Steps 2 and 3, we used (Kalus & Kuhrmann, 2013) to collect contextual criteria as well as our five criteria defined in the Excel add-on.

Step 3 – Analyzing Impact on Process Characteristics: Figure 34 presents the result set of Step 3: As the context factors are now available, Figure 34 (left part) presents their positive (+) and/or negative (-) influences. Furthermore, Figure 34 (right part) also contains the concrete impacts on the improvement goals from our repository (Figure 34, right part).

		6. Context(factors)	Fixed context questions						Individual definable context												
Practices	5. Used Practices (please mark)	unfulfillable Practice Precondition	Teamsize: >10	Team Distribution: onsite	Project Duration: > 6 months	Customer Availability: Daily	User Availability: Daily		1. Fundamental Context	1.1 Self-determination Of Team	1.2 Strict Rollout Plan for SALOG	1.3 Overall Waterfall Process Mod	2. Team Context	2.1 Personnel turnover	2.2 Domain Knowledge	2.3 Tool Knowledge	2.4 Comprehension Of Process	3. Internal Working Environment	3.1 Usage of Prototyping	3.2 IT-QMS	3.3 Management Support
Product Vision Board							+				-				+						
Personas						+	+								+						
Story Mapping				+											+						
Planning meeting				+		+				+	-										
Iteration Reviews				+		+	+			+		-			+						
Backlog grooming						+									+						
Shippable Increment							+				-										
Daily Meeting			-	+						+											
Sign Up				+						+											

Practices	3.6 Project Budget	3.7 Sub-contractors	4. External Working Environment	4.1 Number Of Stakeholders	4.2 Availability Of Stakeholders	4.4 Frequency Of Change Reques	4.6 Type Of Contract	5. Further Conditions	5.5 Safety & Security	5.7 Guidelines For User Interfaces	Not desired
Product Vision Board	+	-		+							
Personas	+			+	+						
Story Mapping	+			-							
Planning meeting	+				+						
Iteration Reviews				-	+						
Backlog grooming	+				+						
Shippable Increment		-							-		
Daily Meeting	+	-									
Sign Up											

Practices	Product quality	Flexibility and ability to change	Customer collaboration	Knowledge transfer	Transparency (of project status)
Product Vision Board	2	2	1,8	1,5	2
Personas	1,62	1	1,55	1,5	1,67
Story Mapping	1,56	1,8	1,71	1,73	1,75
Planning meeting	1,54	0,56	1,5	1,39	1,85
Iteration Reviews	1,47	1,6	1,34	1,41	1,72
Backlog grooming	1,44	1,35	1,63	1,65	1,77
Shippable Increment	1,5	1,36	1,5	1,5	1,91
Daily Meeting	1,43	1,42	1	1,46	1,69
Sign Up	1,4	2	2	0,25	2

Figure 34: Capgemini case study - *Agile Potential Analysis* Step 3

Step 4 – Adapting Agile Practices: Based on this information, the team decided to fill their agile transition backlog with Daily Stand-up, Product Vision Board, Sign-up, and Backlog Grooming (already ordered to their priorities) first. Step 4 was only applied in the case of the Daily Stand-up, where it is common to vary the frequency. In this case, the team decided to hold it twice a week instead of every day. One major reason was the fact that most of them are working only part-time in this project.

As the Capgemini case study was only used as an example for the execution, the other case studies were performed in a similar way and the results (at least the ones with the tabular representation) were presented almost the same as illustrated in the figures above.

5.2.5 Data Collection and Analysis

The data collection method selected for these case studies were questionnaires (Diebold, et al., 2016) because of the expected larger number of participants. The questionnaire was derived from the sub-aspects of the hypotheses, which were refined into questions (presented in Appendix B.1). The structure of the questionnaire was as follows:

Introduction and demographics: The questionnaire included an introduction of the goal and the procedure. Additionally, it encompassed questions for eliciting demographics information about each participant. This was needed for further analysis and comparison across cases and roles and included a unique identifier (per case study) and the role, from the set of “Agile Potential Analysis Executer”, “Process Owner”, and “Process User”.

Goal achievement: Hypothesis H1 is about the appropriateness of the selected improvement actions, in our case the agile practices. Furthermore, it is about the achievement of improvement goals in general as well as about their individual problems, resp. improvement goals. The questions were formulated similar to the others and used a 5-point Likert-like scale. The goal achievement part also included two open questions aimed at eliciting feedback on the goal orientation as well as on the approach in general.

All the following questions except the final open comments were based on commonly used and validated instruments. We used a literature collection of quality criteria for evaluation and their instruments (Guzman, et al., 2017). Based on this input and hypotheses H2 and H3, we identified the criteria for our questionnaire together with two empirical experts. Thus, we were able to use and instantiate a validated question set.

Comprehension: For *understandability*, we referred to (Guzmán Rehbein, kein Datum) for the general purpose and process and to (McKinney, et al., 2002) for the expectation about information quality. (Lee & Strong, 2003) were used to refine *transparency* into relevancy. Finally, the expected *reliability* of the service quality was taken from (Grover, et al., 1996).

Acceptance: All the sub-aspects of acceptance (H3) – *job relevancy*, *results demonstrability*, *output, output quality*, *perceived ease of use*, *perceived usefulness* – are defined by the Technology Acceptance Model (TAM3) (Venkatesh & Bala, 2008).

Closing: At the end, all participants had the opportunity to give feedback by answering an open question. The questionnaire ended with some closing thanks to the participants.

The analysis of the questionnaire is separated into the quantitative analysis and the qualitative analysis. In this thesis, the results of the quantitative analysis and a few qualitative aspects are being reported to explain some of the quantitative results and identify improvement suggestions. The quantitative analysis is based on descriptive statistics for testing the presented hypotheses. This is especially the case for H2 and H3 because they are refined into a number of questions by the validated instruments. The major descriptive value we are using is the median as it is the most robust value on outliers. Since all of the hypotheses will be measured on 5-point Likert-like scales (more details can be found in the next section), the formal hypothesis (and null-hypothesis) are stated as follows:

H_i: Median (item) > 3	*(i = 1, 2, and 3)*
H_i_0: Median (item) = 3	

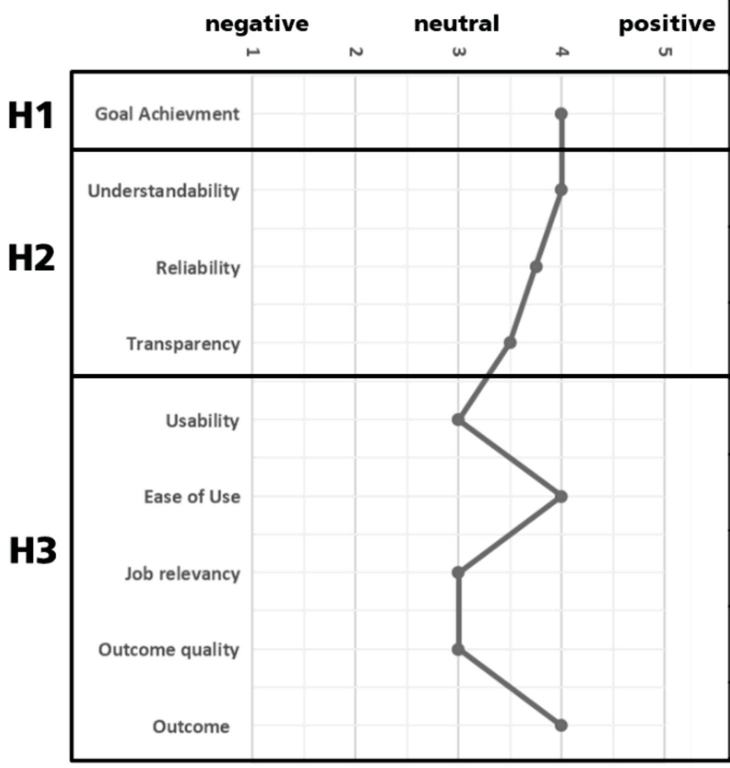
The use of single questions enabled us to aggregate the results of the questions for each cluster or aspect, such as understandability. The *case-specific analysis* was interesting due to the varying contexts and slightly different applications, such as visual vs. tabular result presentation. Besides this, we performed some pooling where the data of all the different case studies was aggregated (similar to the aggregation of the questions for the aspects: the median is built over all 26 individual data sets). This enables an *overall analysis* as well as *role-specific consideration*. The analysis was completed using the overall data for a *correlation analysis* to check the consistency and understand relationship between the different aspects.

5.2.6 Results

This section presents the results of the six case studies. The anonymized raw data can be found in Appendix 6B.2B.2.

5.2.6.1 Capgemini

The case study performed at Capgemini included nine participants, eight team members and one external student. Of the eight team members, one was a process owner and seven were process users, such as developers or experts in user experience. The external student was responsible for performing the *Agile Potential Analysis*, i.e., he assumed the role of the executer.



	Mdn.	Std.Dev.	p	Sig.
Goal Achievment	4	0,486	0,002	✓
Understandability	4	0,768	0,016	✓
Reliability	3,75	0,729	0,043	✓
Transparency	3,5	0,744	0,124	✗
Usability	3	0,678	0,500	✗
Ease of Use	4	0,623	0,007	✓
Job relevancy	3	0,838	0,500	✗
Outcome quality	3	0,535	0,500	✗
Outcome	4	0,791	0,020	✓

Figure 35: Central tendency (median) of the perception of the Capgemini case study participants on goal achievement (H1), comprehension (H2), and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 35 shows the overall perception of the Capgemini participants of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

Regarding the **achievement of the target improvement goals** (H1), the nine participants agreed that the *Agile Potential Analysis* contributes to the achievement of the selected goals (H1) (median = 4, min = 3, max= 5, n = 9). In particular, Capgemini was interested in achieving five goals by introducing agile practices using the *Agile Potential Analysis*, namely product quality, flexibility, customer collaboration, knowledge transfer, and transparency. The participants believed that using the *Agile Potential Analysis* positively contributes to achieving knowledge transfer (median = 4, min = 3, max= 4, n = 7) and transparency (median = 4, min = 3, max= 4, n = 9). Regarding the latter, one participant explained that "this is the case because the first selected agile practice was Daily Stand-up". The other goals were not affected positively nor negatively.

With regard to the **comprehensibility of the selection of agile practices and their impacts** (H2), the participants perceived the *Agile Potential Analysis* as moderately understandable (median = 4, min = 2, max= 4, n = 9), reliable (median = 3.75, min = 2, max= 4, n = 8), and transparent (median = 3.5, min = 2, max= 4, n = 8). The executer rated the understandability, reliability, and transparency of the *Agile Potential Analysis* slightly better than the team members. This might be the case because the executer knew all the details of the analysis process. Furthermore, the results of the process user highly depend on the result presentation, the information it includes, and the reasoning and algorithms of the analysis. In this specific case, the student gave the presentation without any detailed information or reasoning. This fact called our attention to this potential threat when conducting case studies.

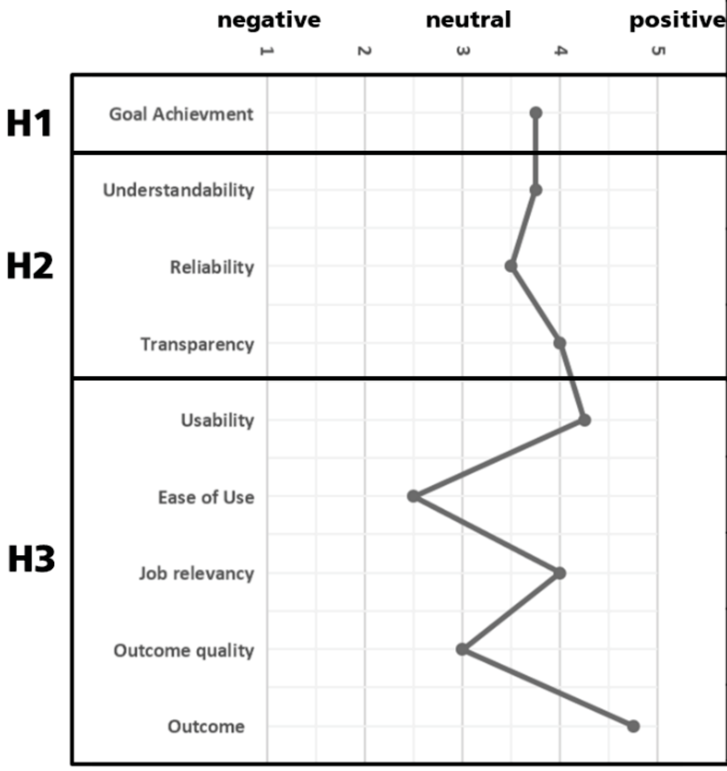
When considering **acceptance** (H3), the participants assessed the *Agile Potential Analysis* as moderately usable (median = 3, min = 2, max= 4, n = 8) and relevant (median = 3, min = 2, max= 4, n = 7) and its results as having fair quality (median = 3, min = 2, max= 3, n = 8). One participant mentioned that he did not rate it as positive because the suggestions (agile practices) were not being implemented at the time being. Furthermore, with regard to job relevancy, the process user mentioned that they were not using the method itself in their job. The participants evaluated the ACAPI approach as easy to use (median = 4, min = 2.5, max= 4, n = 8) and its outcome as positive (median = 4, min = 2, max = 4.5, n = 8).

During the case study, we also asked the participants how the *Agile Potential Analysis* could improve their work, especially with regard to software process improvement. Here they focused on different aspects: "finding aspects that can be improved" is related to the improvement goals, whereas "new techniques to be selected" or "improving my knowledge about applications of agile practices" deals with the suggested practices. They also recommended "to consider different methods and practices and adjust them to the situation". This fits our motivation because on the one hand, there is more than just one specific agile method, e.g. Scrum, and on the other hand, integration of new practices is easily possible in the repository.

5.2.6.2 VSF Experts

In this case study, two employees from the organization participated. One was an internal student who performed the analysis as executer, while the other one was a process user.

Because they were already developing in an agile way, we slightly adapted the analysis: We did not exclude, but rather tagged agile practices already in use to find out whether they were using them correctly.



		Mdn.	Std.Dev.	p	Sig.
H1	Goal Achievment	3,75	0,354	0,076	✗
H2	Understandability	3,75	0,354	0,076	✗
	Reliability	3,5	0	n.a.	n.a.
	Transparency	3,75	0	n.a.	n.a.
H3	Usability	4,25	1,061	0,124	✗
	Ease of Use	2,5	0	n.a.	n.a.
	Job relevancy	4	0	n.a.	n.a.
	Outcome quality	3	0	n.a.	n.a.
	Outcome	4,75	0,354	0,029	✓

Figure 36: Central tendency (median) of the perception of VSF Experts case study participants on goal achievement (H1), comprehension (H2), and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 36 shows the overall perception of the VSF Experts participants of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

Regarding **goal achievement** (H1) the participants agreed on a rather positive contribution (median = 3.75, min = 3.5, max = 4, n = 2). In particular, the VSF Experts participants were interested in eight different improvement goals. Both participants agreed that the contribution of the suggested minimum viable product and planning was positive (median = min = max = 4, n = 2). Transparency and communication (median = 3.5, min = 3, max = 4, n = 2) were also seen as contributing positively to goal achievement, while the others were rated neutral.

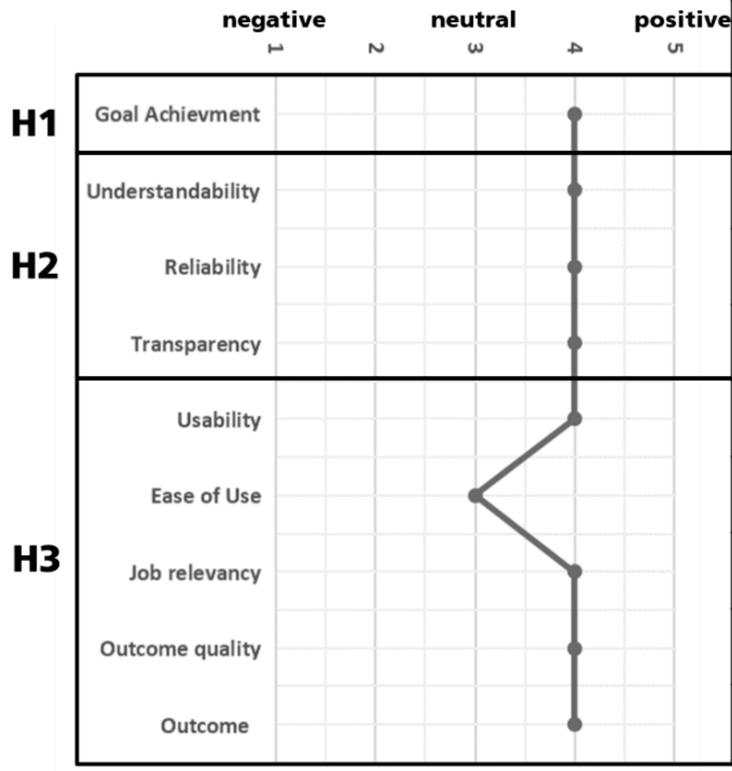
Considering **comprehension** (H2), all three aspects were assessed rather positively. Transparency (median = 4, min = max = 4, n = 2) was rated best, followed by understandability (median =3.75, min = 3.5, max = 4, n =2) and reliability (median = 3.5, min = max = 3.5, n = 2). The participants mentioned that the joint workshop of the customer with Fraunhofer IESE had an influence especially on these aspects.

Regarding **acceptance** (H3) we could see strong differences in the aspects. Outcome (median = 4.75, min = 4.5, max = 5, n = 2), usability (median = 4.25, min = 3.5, max = 5, n = 2), and job relevancy (median = min = max = 4, n = 2) were all three rated with positive agreement, whereas ease of use (median = min = max = 2.5, n =2) was by far rated as the worst aspect. In a short retrospective, both mentioned that "using the model on a whiteboard made it more confusing". The remaining aspect outcome quality (median = min = max = 3, n = 2) was rated neutral.

Both participants mentioned that the suggestion helps them improve their processes, either by "finding and applying proper tools and methods to organize the development phases more efficient" or by "verifying whether a practice is suitable for set goals and given context". The two recommended "conducting the APA - for everyone who wants to be more confronted with new agile methodologies and guided by trusted practices and common workflows/toolsets". They believed using the *Agile Potential Analysis* to be more efficient, especially when someone is completely new to agile development.

5.2.6.3 Kemweb

Kemweb provided seven participants. One of them was the process owner, who also conducted the *Agile Potential Analysis*. As CTO of the company and certified Scrum Master, he had the highest experience in agile development in this company. The other six participants were process users from different disciplines, e.g., project management, design, and development.



		Mdn.	Std.Dev.	p	Sig.
H1	Goal Achievment	4	0,476	0,003	✓
H2	Understandability	4	0,567	0,006	✓
H2	Reliability	4	0,488	0,003	✓
H2	Transparency	4	0,345	0	✓
H3	Usability	4	0,5	0,004	✓
H3	Ease of Use	3	0,476	0,5	✗
H3	Job relevancy	4	0,488	0,003	✓
H3	Outcome quality	4	0,535	0,005	✓
H3	Outcome	4	0,732	0,017	✓

Figure 37: Central tendency (median) of the perception of the Kemweb case study participants on goal achievement (H1), comprehension (H2), and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 37 shows the overall perception of the Kemweb participants of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

With a small difference, all participants agreed on good **goal achievement** (H1) using the presented analysis (median = 4, min = 3.5, max = 5, n =7). The achievement of four of their six initially selected improvement goals – project transparency (median = 4, min = 3, max = 5, n = 5), translation of wishes to requirements (median = 4, min = 3, max = 5, n = 6), communication (median = 4, min = 3, max = 5, n = 5), and knowledge transfer (median = 4, min = 3, max = 4, n = 5) – was rated as positive. The perception regarding the other two goals, understandability of the customer and documentation, was neutral. With the exception of the goals transparency, translation, and communication, the process owner (who was also the executer) did not differ from the process owner.

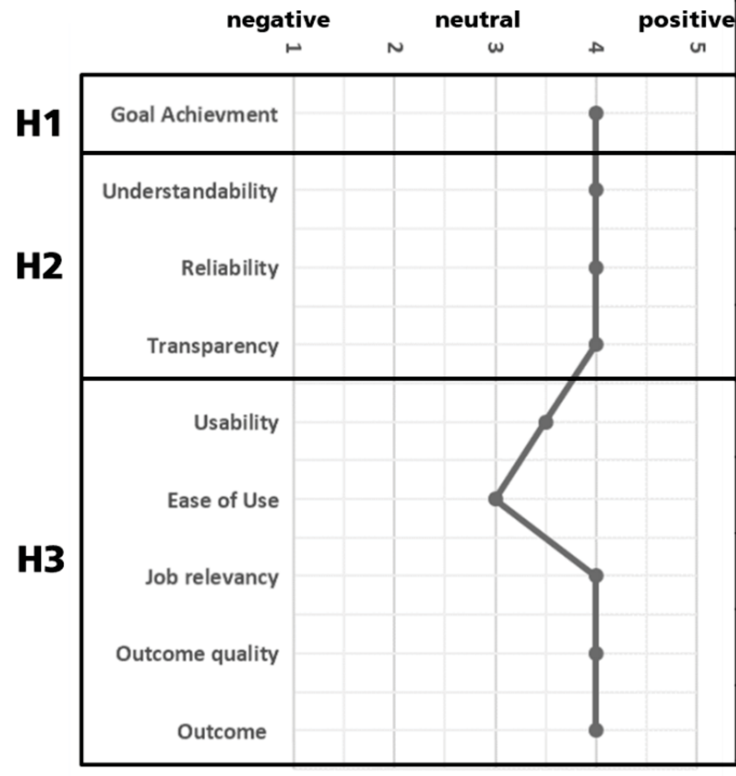
With regard to the **comprehension** of the *Agile Potential Analysis* (H2), all the participants rated its understandability (median = 4, min = 3, max = 4.5, n=7), reliability (median = 4, min = 4, max = 5, n=7), and transparency (median = 4, min = 3.5, max = 4.5, n=7) as positive. From the role perspective, the process owner rated reliability (median =5) and transparency (median 4.5) slightly better than the other participants did. He explained this by "knowing the analysis best as executer and having the highest agile knowledge in the company".

Considering **acceptance** (H3), only ease of use (median = 3, min = 2, max = 3.5, n = 7) was rated negative. In this case, the participants had only been given the presentation without detailed explanations due to time constraints. The executer with the best insights into the method was the participant who provided the lowest-ranked value. All the other aspects – job relevancy (median = 4, min = 3, max = 4, n =7), outcome (median = 4, min = 3, max = 4, n =7), and quality (median = 4, min = 3, max = 5, n =7) were considered as positive.

In addition, the aspect that by using the *Agile Potential Analysis*, "the potential for an agile process can be captured and introduced in the daily work of a team", concrete improvements were mentioned by the participants, such as "optimization of knowledge transfer".

5.2.6.4 TQsoft

Five employees of TQsoft took part in their case study. The participating CEO also holds the process owner role, whereas the other four were process users with different foci, e.g., developer, tester, or project manager. More details about this can be found in (Diebold, et al., 2018).



	Mdn.	Std.Dev.	p	Sig.
Goal Achievment	4	0	n.a.	n.a.
Understandability	4	0,742	0,027	✓
Reliability	4	0,548	0,011	✓
Transparency	4	0,224	0	✓
Usability	3,5	0,418	0,045	✓
Ease of Use	3	0,548	0,5	✗
Job relevancy	4	0,894	0,045	✓
Outcome quality	4	0,548	0,011	✓
Outcome	4	0,908	0,046	✓

Figure 38: Central tendency (median) of the perception of the TQsoft case study participants on goal achievement (H1), comprehension (H2), and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 38 shows the overall perception of the TQsoft participants of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

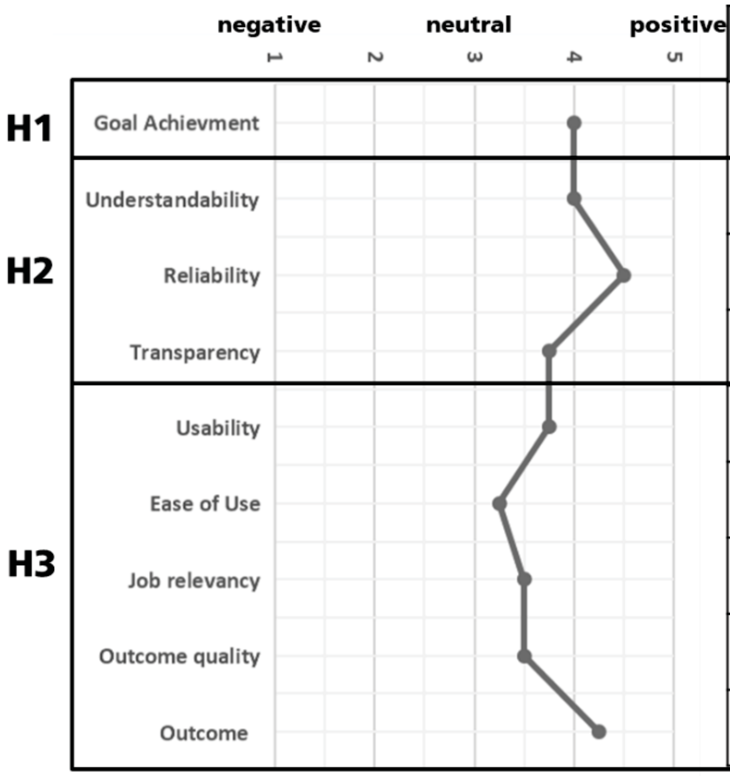
With respect to the **achievement of improvement goals** (H1), all participants agreed on a positive effect (median = min = max = 4, n = 5). Nevertheless, of the four improvement goals – documentation, higher QA effort, improved project approval, and software architecture – only documentation was rated positive (median = 4, min = 3, max = 5, n = 5). The other three were rated neutral.

Regarding the **comprehension** of the *Agile Potential Analysis* (H2), all three aspects, i.e., understandability (median = 4, min = 3, max = 5, n = 5), reliability (median = 4, min = 3.5, max = 5, n = 5), and transparency (median = 4, min = 4, max = 4.5, n = 5) were seen positive. Here the participants mentioned that "it positively influenced the results that the executer from Fraunhofer IESE briefly presented the results".

Compared to comprehension, **acceptance** (H3) had different results for the various. Especially ease of use (median = 3, min = 2, max = 3.5, n = 5, not significant) and usability (median = 3.5, min = 3, max = 4, n = 5) were rated neutral or rather neutral. The other three, i.e., job relevancy (median = 4, min = 2, max = 4, n = 5), outcome (median = 4, min = 2.5, max = 5, n = 5), and outcome quality (median = 4, min = 3, max = 4, n = 5), show the positive aspects of acceptance.

5.2.6.5 Yatta

In the case study of Yatta, only two people participated. One of them was a process owner, called project lead in their individual terminology. The other one, a developer, was classified as a process user. Both of them already had experience with agile development, especially since they were using some Scrum-like processes in most of their projects.



		Mdn.	Std.Dev.	p	Sig.
H1	Goal Achievment	4	0	n.a.	n.a.
H2	Understandability	4	1,414	0,196	✗
	Reliability	4,5	0,707	0,07	✗
	Transparency	3,75	1,061	0,201	✗
H3	Usability	3,75	1,061	0,201	✗
	Ease of Use	3,25	1,768	0,421	✗
	Job relevancy	3,5	0,707	0,206	✗
	Outcome quality	3,5	2,121	0,368	✗
	Outcome	4,25	0,354	0,043	✓

Figure 39: Central tendency (median) of the perception of the Yatta case study participants on goal achievement (H1), comprehension (H2), and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 39 shows the overall perception of the Yatta participants of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

Goal achievement (H1) in general was seen as very good (median = min = max = 4, n = 2), especially because all four mentioned improvement goals – time to market (median = min = max = 4, n = 1), receiving feedback (median = min = max = 5, n = 1), vague requirements (median = min = max = 5, n = 1), and knowledge transfer (median = min = max = 5, n = 2), were rated positive. Especially with regard to goal feedback, one employee mentioned that "involving customers and their feedback as soon as possible leads to results that the customer needs".

With respect to the **comprehension** of the *Agile Potential Analysis* (H2), all three aspects were seen as positive. Reliability (median = 4.5, min = 4, max = 5, n = 2) was rated best, followed by understandability (median = 4, min = 3, max = 5, n = 2) and transparency (median = 3.75, min = 3, max = 4.5, n = 2). Both participants added that existing knowledge of agile development influences the assessment, especially of reliability.

Regarding the **acceptance** of this analysis method (H3), they agreed especially on the valuable outcome (median = 4.25, min = 4, max = 4.5, n = 2) with different viewpoints on quality (median = 3.5, min = 2, max = 5, n = 2). The other aspects usability (median = 3.75, min = 4, max = 4.5, n = 2), ease of use (median = 3.5, min = 2, max = 4.5, n = 2), and job relevancy (median = 3.5, min = 3, max = 4, n = 2)were rated slightly positive.

5.2.6.6 Bosch

In the Bosch case study, only one person from their team, of which three were involved in the execution, participated in the study. Although this person was responsible for the agile transformation of this new organizational unit, he classified himself as a process user.

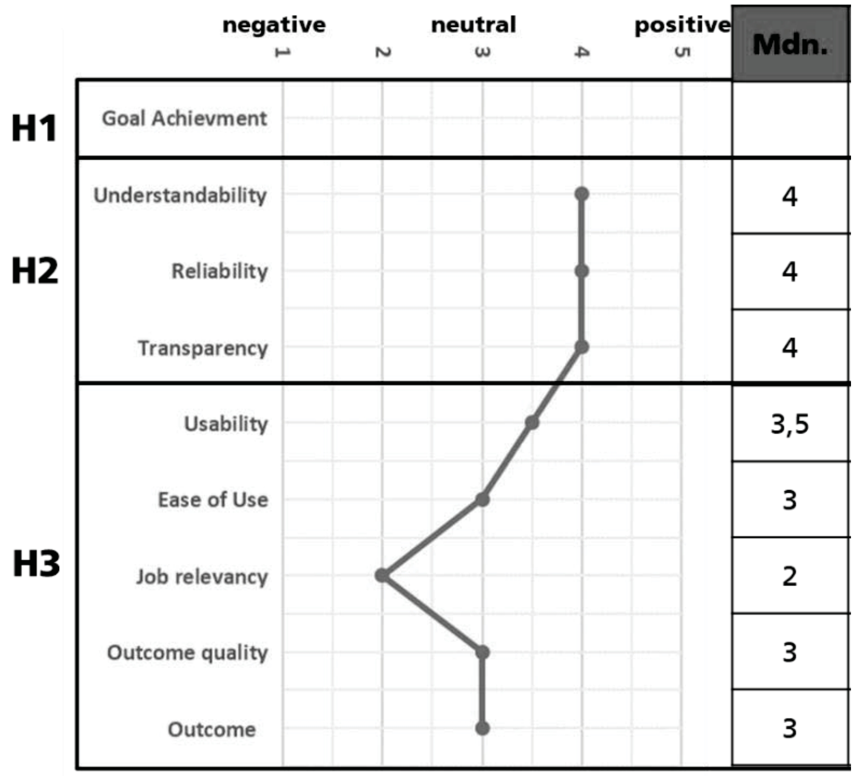


Figure 40: Central tendency (median) of the perception of the Bosch case study participant on comprehension (H2) and acceptance (H3) (5-point Likert scale from negative 1 to positive 5; $\alpha = 0.05$)

Figure 40 shows the perception of the Bosch participant of the *Agile Potential Analysis* with respect to goal achievement, comprehension, and acceptance.

The only participant of this case did not answer the questions related to **goal achievement** (H1), answering neither the generic questions nor for the individual goals regarding product quality, flexibility, customer collaboration, knowledge transfer, and transparency.

With respect to the **comprehension** of the *Agile Potential Analysis* (H2), all three aspects – reliability, transparency, and understandability – were rated positive (median = 4).

Considering **the acceptance** of the approach (H3), the aspects vary among each other and were rated worse. The only slightly positive one was usability (median = 3.5), whereas ease of use, outcome, and outcome quality (all median = 3) were considered as neutral. For the single participant, the *Agile Potential Analysis* was not relevant for his job (median = 2).

5.2.7 Comparison and Interpretation

In this section, different comparisons will be used to arrive at the best generic interpretations for the results in order to draw some conclusions. We will start by comparing the results of the different case studies presented above. The single cases will also be compared with the pooled data aggregating the results of all the case studies. This will be followed by a role-specific analysis. Finally, a correlation analysis will be performed, again using the pooled data.

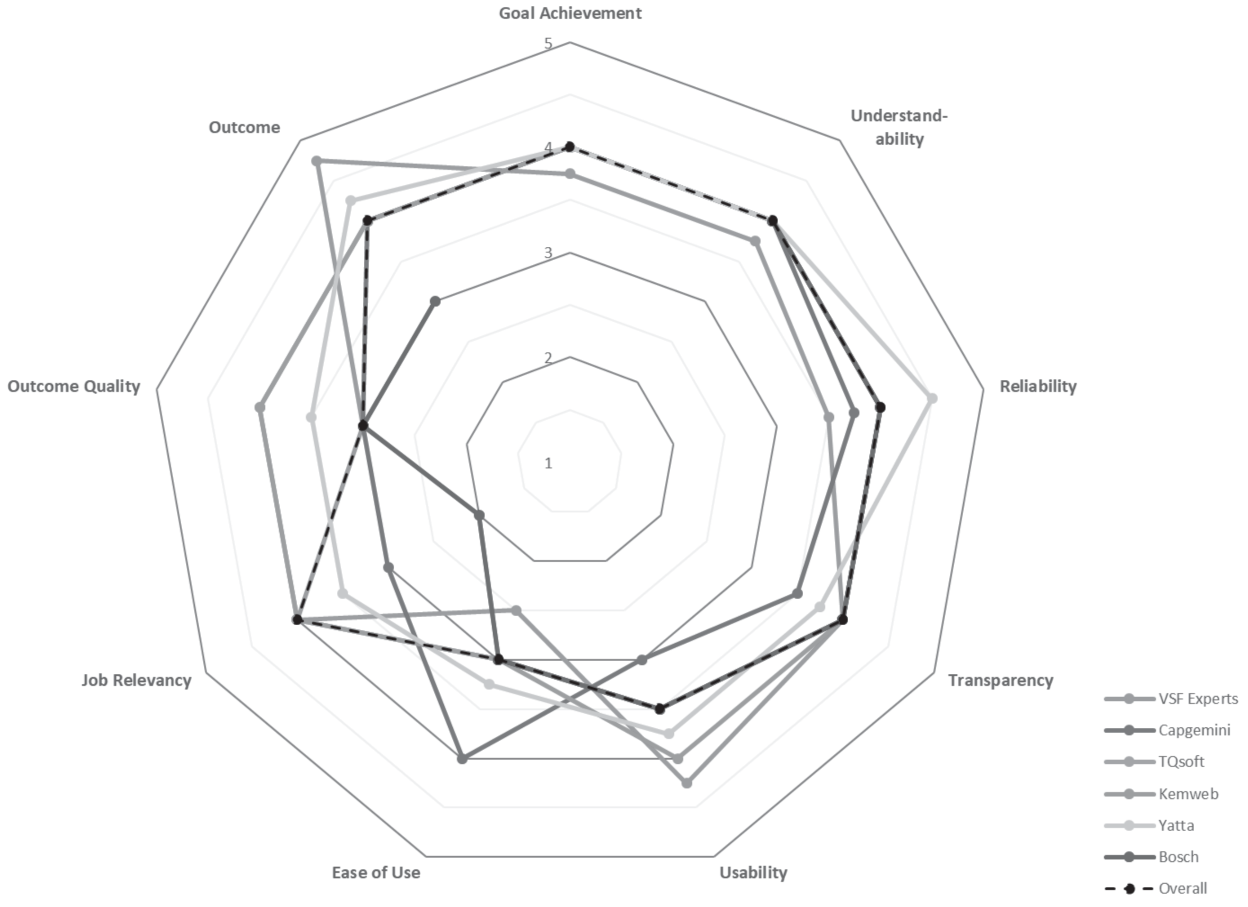


Figure 41: Comparison of *Agile Potential Analysis* case study results

In total, Figure 41 shows that there were almost no negative results (less than 3), and that these were only for two sub-aspects, job relevance in the Bosch case and ease of use in the VSF Experts case. There were some neutral results (3), but most of the results turned out to be positive: When the overall value is calculated with the median (Figure 41, black-dashed line), all aspects except for outcome quality and ease of use (3) and usability (3.5) have a good agreement value (4).

When comparing the results of the different case studies, goal achievement as well as understandability are the aspects with the least variation and thus exhibit the strongest agreement among the cases (3.75 – 4). This is followed closely by transparency (3.5 – 4). The highest range is observed in the outcome between the VSF Experts case and the Bosch case (3 – 4.75). Nonetheless, these ranges need to be considered with care due to the fact that in the Bosch case, for example, which has two negative outliers (outcome and job relevancy) only one participant answered, meaning that the results are not significant and thus not as trustworthy.

When we aggregate these results of the aspects with our three hypotheses, hypothesis H1 on goal achievement and H2 on comprehension (refined by understandability, reliability, and transparency) are supported. This is not the case for H3 on acceptance (refined by the remaining aspects) because the different aspects vary from their rated values.

To make this comparison clearer, Table 19 presents an overview of the statistical values.

Table 19: Comparison of the medians of the cases (values marked with * are not statistically significant)

Hypothesis / Aspect	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Aggregated
Goal Achievement (H1)	**4**	**3.75***	**4**	**4**	**4***	**-**	**4**
Comprehension (H2)	**3,75**	**3,75**	**4**	**4**	**4**	**4**	**4**
Understandability	4	3.75*	4	4	4*	4*	4
Reliability	3.75	3.5*	4	4	4.5*	4*	4
Transparency	3.5	4*	4	4	3.75*	4*	4
Acceptance (H3)	**3**	**4**	**4**	**4**	**3,5**	**3**	**3,5**
Usability	3*	4.25*	4	3.5	3.75*	3.5*	3,5
Ease of Use	4	2.5*	3*	3*	3.25*	3*	3
Job Relevancy	3*	4*	4	4	3.5*	2*	4
Outcome Quality	3*	3*	4	4	3.5*	3*	3
Outcome	4	4.75	4	4	4.25	3*	4

Next, we will not only consider the results case-by-case, as in the previous paragraphs, but we will also check whether the participating roles might have had an influence on our results. Compared to four process owners and three APA executers, the 20 process users were the largest group. Figure 42 shows that there are some differences among the roles. The highest differences are that the APA executers rated usability higher than the process users or owners. Furthermore, considering the reliability of the *Agile Potential Analysis* , the results of the process owners were higher than those of the APA executers and process users.

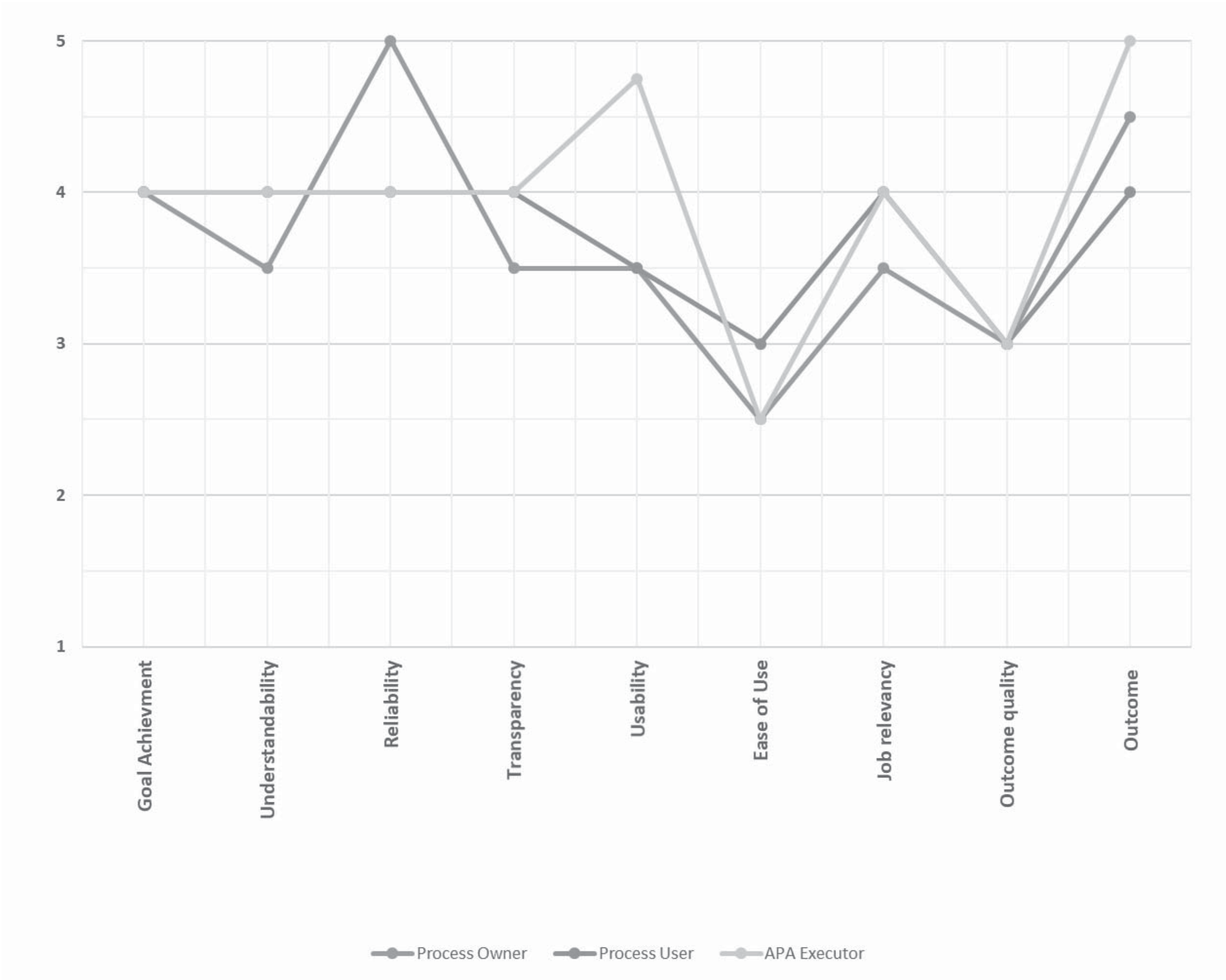


Figure 42: Role-specific comparison of *Agile Potential Analysis* case study results

Having shown the results among the different cases as well as the different roles, in the following we will give our main interpretations of these results:

- Goal achievement was rated lower in the VSF Experts case because of their two different foci: One focus was from the viewpoint of the product owners, and the other from the remaining team members (including the Scrum Master).

- Reliability was rated higher in the case of the process owners because of their process awareness as well as process and agile knowledge.

- Transparency was rated lower in the Capgemini case because their result presentation had less details about the *Agile Potential Analysis* and was given by a novice student.

- Usability was rated highest of all in the VSF Experts case because of the strong involvement of Fraunhofer IESE in Steps 2 & 3: We performed both steps together in a workshop on a whiteboard using the model.

- Usability was rated higher in the case of APA executers[14] because they know the method best and in its individual details.

- Ease of use was rated higher in the Capgemini case because of their company size. Usually, larger companies have dedicated software process improvement initiatives, into which our analysis fits easily.

- Ease of use was rated lower in the VSF Experts case because they were the (first and) only ones using the complex model representation.

- Outcome was rated higher if the role was more involved in the execution of the *Agile Potential Analysis*. Usually the executer is more involved than the process owner, who is even more involved than a regular process user. This ordering is also the ordering of their outcome rating.

In addition to these comparison results and interpretations, we performed a correlation analysis to obtain more insights. Although we did correlate all the single questions or aspects, in the following only the correlation of the sub-aspects of the hypotheses will be presented and discussed briefly, e.g., understandability, reliability, etc. However, an important aspect that emerged and increased the overall reliability of the instrument was the

[14] Excluding Fraunhofer IESE employees as executers (at least from filling out the survey and thus also from the results)

fact that the correlation analysis at the question level showed high positive correlation within an aspect, such as outcome quality. Table 20 depicts the correlation matrix.

Table 20: Correlation matrix (aggregated over all the different cases)

	Goal Achievement	Understandability	Reliability	Transparency	Usability	Ease of Use	Job Relevancy	Outcome Quality	Outcome
Goal Achievement	1.00								
Understandability	0.45	1.00							
Reliability	0.35	0.41	1.00						
Transparency	**0.74**	**0.71**	0.36	1.00					
Usability	0.48	0.50	0.40	0.66	1.00				
Ease of Use	-0.11	0.40	-0.06	0.11	-0.18	1.00			
Job Relevancy	0.30	0.62	0.44	0.56	0.67	-0.12	1.00		
Outcome Quality	0.43	0.44	0.40	0.63	0.39	0.20	0.30	1.00	
Outcome	0.16	0.42	0.67	0.30	0.32	0.00	0.40	0.24	1.00

Transparency and goal achievement as well as transparency and understandability are the two pairs of aspects with high positive correlation (> 0.7). Table 20 also shows that there are some other pairs of aspects that almost have high agreement: usability and transparency, job relevancy and usability, job relevancy and understandability, outcome quality and transparency, as well as outcome and reliability. The correlation results also show only a few negative results with a very weak negative correlation. Nonetheless, it is interesting to see that three of these four are combinations with ease of use, one of the categories with a large range of answers.

5.2.8 Threats to Validity

Threats to the validity of this validation will be discussed along four dimensions: construct, internal, conclusion, and external validity (according to (Wohlin, et al., 2012)):

- *Construct validity* is defined as the degree to which the variables used accurately reflect the constructs of interest.
- *Internal validity* is defined as the degree to which conclusions can be drawn about the effect of the independent variables (e.g., usefulness and ease of use) on the dependent variables.
- *Conclusion validity* is defined as the degree to which the results of the research are statistically significant.
- *External validity* is defined as the degree to which the results of the research can be generalized.

In this validation, **construction validity** mainly regards mathematical and statistical aspects. To test the given hypotheses from a statistical point of view, we used the common T-test (Student, 1908). With this test and the given p-value within it, we were able to show the statistical relevance of most of the results.

With regard to **internal validity**, conducting this validation as industrial case studies made it impossible to control effects related to history, maturation, statistical regression, or mortality (Wohlin, et al., 2012). Furthermore, the different case studies did not necessarily have the same improvement goals because their selection or identification is part of the approach itself.

Since the *Agile Potential Analysis* fosters the separation between the coach or expert in the method and the participant, internal validity increases, because in the least separated manner, the expert is conducting the elicitation steps, e.g., workshops, and presents the results at the end.

Since all the study subjects performed the approach for the first time and not a second time (so far at least), there is no specific learning effect. Nonetheless, it might be the case that the participants' experience with regard to agility or specific agile practices might have differed such that there might have been some kind of learning effect.

Finally, depending on the analysis itself, the output, which is the transition backlog containing the agile practices, is sometimes too large, which makes it hard to read. We will further discuss this in the *Simulation of Process Improvements* as well as in the future work section since there is a trade-off between level of detail and understandability.

Considering **conclusion validity**, the most important aspect is the creation of the validated and reliable instrument. All questions for H2 and H3 are validated and reliable because they are based on existing validated instruments such as TAM (Venkatesh & Bala, 2008) or other instruments mentioned above. In addition, this instrument as well as the evaluation approach were peer-reviewed by several experts. Considering the data analysis, the pooling of the data needs to be discussed, especially as the companies in our case studies differed so much. The four SME cases are quite similar, so comparing them is a valid option. Furthermore, since all our case studies covered single teams and not larger projects, it makes sense to combine them in one pool in order to get a more reliable statistical analysis.

External validity is about threats that we could not really mitigate: On the one hand, we had the different maturities and organizational contexts of the participating companies, from small SMEs to large enterprises, from domains ranging from information systems to embedded software and systems development. This issue was already discussed above in the context of conclusion validity. However, three of the six cases also had a relationship with Fraunhofer IESE by working on a joint research project. All in all, our total of 26 participants already provides a good indication for generalizing the results, especially for SMEs, because we had many participants from SMEs.

On the other hand, as stated in the execution section (cf. Section 5.2.3), the execution and duration of the *Agile Potential Analysis* was slightly varied due to internal learning on the method as well as due to the specific contexts. Furthermore, the fact that the questionnaire was sent to some participating companies directly after the analysis and to others when they had already started the transition might have had an impact on the results.

5.3 Validation of Simulation of Process Improvements

This section presents the concrete validation for the *Simulation of Process Improvements*. It covers all aspects from the research design to the data analysis and interpretation.

5.3.1 Research Design

During the validation of the *Simulation of Process Improvements*, it is not necessary to cover the first three steps of Figure 32. Due to the fact that not all participants and companies from the validation of the *Agile Potential Analysis* took part, we needed to perform the “case selection” again (cf. Figure 32).

Simulation: The main part is the execution of the simulation, which was performed as described in Section 4.4.

Feedback: According to the same procedure used for the *Agile Potential Analysis*, the execution was followed by data collection. Considering the same hypotheses, the collection was done using semi-structured interviews.

After presenting the study procedure, and in order to refine the overall validation methodology, it is important to mention that this validation contains one single case study covering several participants from different companies using the same scenario. This is one of the major differences to the research design of the *Agile Potential Analysis* validation.

The other important difference is that we only considered the hypotheses regarding comprehension (H2) and acceptance (H3) as important. The goal achievement hypothesis (H1) is not important because the simulation is only built on the selected goals of the analysis.

5.3.2 Population and Sample

In general, the population for the validation of the *Simulation of Process Improvements* was the same as for the *Agile Potential Analysis* because it required the same experience in software engineering as well as agile development. Furthermore, having a theoretical background in mathematics, statistics, and simulation is not necessary but helps to understand the creation of some of the results during the *Simulation of Process Improvements*.

As we had six case studies for the validation of the first part of the overall method, the *Agile Potential Analysis*, the idea was to motivate as many people as possible who participated in the first validation to also participate in the second validation. Since our focus was on validating whether the *Simulation of Process Improvements* brings additional benefits, these people were the best candidates, as they already knew the analysis method. Four people agreed to participate: With two from TQsoft, one from Kemweb, and another one from Capgemini, we could ensure at least some variation (cf. Table 21).

Table 21: Participant profiles

Company	Context	Participant's Role	Organizational Role
Capgemini	Enterprise	Process Owner	Project Manager
TQsoft (TQ1)	SME	Process User	Project Manager
TQsoft (TQ2)	SME	Process Owner	CEO
Kemweb	SME	Process Owner, APA Executer	CTO

5.3.3 Execution

Due to the tremendous amount of effort that is necessary to perform the complete *Simulation of Process Improvements* for each company individually, we decided to validate it in a lightweight way together with a fictitious example inspired by real cases. Thus, we created a single scenario, which was used as part of a walkthrough (Wharton, et al., 1994).

Even if the scenario was the same for all participants, it was introduced by starting with the individual results of the *Agile Potential Analysis*, which led to the scenario of the simulation. Since the scenario was provided as a slide set (PDF) that only presented the results of the different steps and hid most of the theoretical aspects, the walkthrough of the scenario was performed via telephone. Because the results of the first two steps (Steps 1 & 2) and the following two steps (Steps 3 & 4) look similar, our scenario only contained the first two steps. Nevertheless, we explained that the customer-specific results would look similar regarding its presentation. During the walkthrough, we allowed questions, which we recorded for later analysis. After all necessary clarifications, we proceeded with the data collection in the phone calls.

5.3.4 Data Collection and Analysis

The data collection method for this case study were interviews (Diebold, et al., 2016) because of the smaller number of participants and because we already presented the scenario via telephone anyway. In combination with the walkthrough of the example scenario, each phone call took 30 to 35 minutes, of which the interview took about 20 minutes on average.

The semi-structured interviews were conducted based on a predefined interview guidelines covering the necessary topics. Since our original idea was to reuse the questionnaire from the *Agile Potential Analysis*, which was based on validated questions, we at least based the interview guidelines on these questions. They were instantiated for the *Simulation of Process Improvements* and partially rephrased as open questions, as appropriate for interviews. After a short introduction, the following sections from the interview guidelines were used:

- General impression compared to the *Agile Potential Analysis*
- Comprehension
- Acceptance
- Further feedback

Especially the sections on comprehension and acceptance were refined by one aggregated open question for each of the aspects that we had used for the *Agile Potential Analysis*. The detailed interview guidelines with all questions are attached in Appendix B.1.

Because the data collection was different from the validation of the *Agile Potential Analysis*, the analysis was also conducted in a different way. The qualitative nature with the open interview questions led to a data-driven thematic analysis with coding aspects. We performed the coding according to hypotheses H2 and H3 as well as the value of the simulation in general. Especially the latter aspect was meant to replace the non-applicability of hypothesis H1 because its execution was not case-specific and addressed the same improvement goals as the *Agile Potential Analysis*.

5.3.5 Results

The following subsections present the results of the single interviews and address their differences, e.g., with regard to context (cf. Section 5.2.2 and 5.3.2). All of them are structured according to the parts mentioned in the data collection section. The raw data can be found in Appendix 6B.2B.3.

5.3.5.1 Capgemini employee

The first impression mentioned by this participant was “great” in comparison with the *Agile Potential Analysis* and its results. This overall initial statement could then be further explained and refined when we started with the detailed aspects.

Having the *Simulation of Process Improvements* in addition to the *Agile Potential Analysis* increases understandability by “seeing directly the implication” and “how the results come about”. Furthermore, this interviewee took the view that the visualizations are understandable with the explanations of the different steps performed. These results correlate with the higher transparence than the *Agile Potential Analysis*. Our participant mentioned that “even without knowing and seeing the mathematical details, it is visible what is calculated” such that it is possible to do one’s own comparisons and reprioritize the different improvement goals and agile practices. He also believed that “the Agile Potential Analysis wet-nurses the company too much”. Reliability as the last aspect of comprehension was something else he considered improved, due to better traceability of how the results are achieved (“if I only get results, I don’t trust them per se”) and the possibility to make your own decisions.

Compared to the *Agile Potential Analysis*, acceptance only increased slightly because “interpretation is obviously still necessary”. Furthermore, acceptance in general strongly depends on how the results are presented to the team, project, or organization. With a “presentation, as given in this case study, and the walk-through it strongly increases acceptance but without there would be no acceptance".

In general, the participant believed that "this add-on is a development in the right direction and a super update". Furthermore, he would like to take the two parts of the complete method as a basis for discussions with other people within his organization in order to spread the method and the results.

5.3.5.2 Kemweb employee

Compared with the *Agile Potential Analysis*, this interviewee stated that "it is the next logical step with the evaluation and weighting with existing data" and thus brings benefits to the overall approach. Even if he found the simulation in general quite good, he mentioned that he "did not get the calculations in detail, but probably I do not need to". The following detailed responses regarding comprehension and acceptance of the simulation reveal more insights into his overall evaluation.

Initially, the participant had problems assessing understandability due to the complex figures presenting the results. He mentioned that "an explanation is necessary" since there is more content than before. Most important with regard to this issue is the fact that the interviewee mentioned that "it is worth investing the time for understanding this add-on". Furthermore, due to the increased level of details, transparency was rated good and increased compared to the *Agile Potential Analysis*. The final sub-aspect of comprehension was reliability. Due to the character of a suggestion of the simulation, the participant mentioned that the results are not more or less reliable but more detailed and precise.

The details that are seen also increase acceptance because now the "weak spots (in our case the negatively influencing agile practices) are easier visible". Furthermore, the scientific as well as mathematical foundation underlying this simulation approach helps to increase its acceptance. Finally, "the transfer of understanding is moved from the respective case company to the method". Especially this last aspect is very interesting because not all participants of the *Agile Potential Analysis* case studies would like to get this transfer taken off from them. Nevertheless, for this participant it was a benefit. All these aspects influencing acceptance were not only true for the results of the simulation, but also for the quality of these results.

5.3.5.3 TQsoft employees

Even though this was already mentioned in the population section, it is important to remind the reader again that we got two participants from this case company taking part in this case study in separate interviews.

From the general perspective, the second participant (TQ2) mentioned being "overwhelmed with content such that explanations are necessary and interpretation needs to be learned". Thus, he also started thinking about how to narrow the content down.

Understanding as a sub-aspect of comprehension was considered improved by the visualization, even if "not everybody likes visualizations more than textual information". Although one participant (TQ1) mentioned that the reduced information helps, the other one (TQ2) felt that "much information was put on one slide". Some minor issues regarding the visualization were mentioned, such as diagrams with the value "0.0". One interviewee (TQ1) stated that "the goal needs to create trust", which is better shown in visualizations than in (Excel) lists. This is the case because "in Excel lists it would be easy to add some numbers" and "visualizations are the cure", as they create a feeling of a lot more possibilities. From the transparency perspective, only the first employee (TQ1) stated that "it got slightly better but could still be improved".

The acceptance of the results as well as their quality was answered together by both interviewees. Both agreed that the simulation "impresses more and thus appears more sustainable”. Nevertheless, the project manager (TQ1) mentioned that this “might be seen differently by different organizational roles”, e.g., it might be too abstract for a developer.

In the section about further feedback, these two participants were quite creative in finding new ideas and input for further improvements and features of the simulation, particularly its representation. The one (TQ1) who mentioned that the slides were too full stated that “a more interactive step-by-step build-up with more details and icons would further increase the understanding” and transparency. The other one (TQ2) focused on four aspects: First, he (TQ2) mentioned that involving the audience earlier still remains an issue to be improved. Second, he was wondering about the agile practices already in use and mentioned that only “tagging” them would be better than deleting them. In the context of this suggestion made during the interview, we also discussed the issue of being flooded with information. Third, having this visualization, this interviewee saw the possibility of using the range between worst- and best-case scenarios to show more potential even in practices that might already be used and could be improved. Finally, different aspects of visualizations were suggested, e.g., “not showing all four categories but aggregating them”.

5.3.6 Comparison and Interpretation

After considering the three different runs of this case study, we will aggregate and interpret them based on the different hypotheses or categories. Table 22 shows these aggregated results.

Table 22: Aggregated comparison of results of the validation of the *Simulation of Process Improvements* (+ = positive aspects; o = neutral aspects; - = negative aspects)

Category	Capgemini employee	Kemweb employee	TQsoft employees
Value in general	+ "First impression: very good"	+ "Add-on is an added value" + "Find it very good" + "Didn't get details of calculation"	+ New feature: risk treatment – "First impression: too much; explanation necessary"
Comprehension (H2)	+ "Effects can be seen directly" + "Visualization is understandable" + "Much more transparent" + "Concrete comparison is now possible" + "Very reliable"	– Understandability of the graphic is quite complex – "Explanation is necessary" + "Additional value is worth spending explanation time" + "Level of detail increases transparency" + "More detailed and precise"	+ "Visual representation is simpler because of reduction" + "Representation shows possible improvements" o Interactive steps would increase comprehension + "Visualization is much superior to the list" + "Visualization seems to give more opportunities"
Acceptance (H3)	+ "Presentation with explanation has much stronger acceptance" (without, not) o "More mathematical details not necessary"	o "Quite nice", shows drawbacks of practices directly o "Take over transfer by visual representation"	+ "Seeing the effect directly increases acceptance" + Impression of visualization leads to more sustainability in the results + Acceptance improved but this might not hold for every role
Further information	+ "Great update" + Good basis for discussing and promoting the approach to others		+ "Interactivity in Excel as well as in visualization would help" + Showing further potential

All the different participants from the three participating companies agreed in general that the *Simulation of Process Improvements* is a very good add-on to the *Agile Potential Analysis*, although they could not agree from their first impression on whether too much content and explanation is necessary. Whether and how these general results are the same or agreed on among the participants for the different hypotheses will be discussed in the following:

Comprehension (H2), which is described by understandability, transparency, and reliability, showed diverse results. Nonetheless, all agreed that the aspect of graphical visualization improves all mentioned aspects. The detailed level of granularity enables concrete comparisons. Furthermore, some agreed that with too many details or too much content, the representation gets too complex and needs a lot of explanation. From this point of view, we identified that it is important to find the right trade-off between the visualization and the detail of content to be presented.

Regarding acceptance (H3), there was strong agreement among the participants that the *Simulation of Process Improvements* part increases the acceptance of the overall approach. This is especially the case because more details such as drawbacks of practices or direct effects can be seen in the visual representation. The only issue discussed regarding acceptance was the aspect of how much mathematical and statistical background is meaningful or necessary to show to the companies where the method is performed. Here they could not agree because some were more interested than others, who were happy to just see results they can trust.

Independent of the two hypotheses we covered by this case study, we were able to collect further feedback and information from the case study participants. All of them were quite happy, similar to their first impression, and all of them provided completely different ideas of how to provide additional information or provide the information in a better way, or how to use this method and its results for further purposes. These possible new features and further ideas for using our approach will be discussed in the future work section at the end of this thesis.

Summarizing the results of this case study on the *Simulation of Process Improvements* part and its interpretations, we conclude that although the *Agile Potential Analysis* could be performed as a stand-alone approach, the *Simulation of Process Improvements* seems to be an additional improvement regarding both of the analyzed hypotheses, comprehension and acceptance, as it brings several benefits to its customers or users. Furthermore, with some creativity, it would be possible to find even more meaningful applications than those for which the simulation is used for in its current form.

5.3.7 Threats to Validity

Just like we did for the first validation, we will discuss the threats to validity along four dimensions: construct, internal, conclusion, and external validity.

Construct validity comprises two major threats. First, all the interviews we conducted were validated by giving the notes to the interviewees for verification. Second, we are aware of the fact that all the results and interpretations of this study can only be treated as initial indications. Compared to the validation of the first part, this is the case because we had only one case study with a smaller number of participants.

Internal validity comprises the largest number of threats. First of all, this evaluation was also conducted as industrial case study, meaning it is not possible to control effects related to history, maturation, statistical regression, or mortality (Wohlin, et al., 2012).

In contrast to the *Agile Potential Analysis*, conducting the *Simulation of Process Improvements* is much more difficult and time-consuming. Because we needed the knowledge and opinions of experts, who do not have much time, we conducted a walk-through during the interview moderated by the method creator. Furthermore, as we used only one example scenario in all the interviews, the threat of individual goals present in the first validation does not exist here because all had the same scenario with given goals.

Regarding the selection of the participants, we decided to pick the participants of the validation of the *Agile Potential Analysis* because of their knowledge about this part of the overall approach. There was no learning effect because they applied the add-on to the *Agile Potential Analysis* only once.

As already stated by some of the study participants, the complexity of the visualization is a potential threat to internal validity because showing and visualizing so much information might confuse them and thus would also influence the results of the study.

As for **conclusion validity**, the main issue is the data collection instrument we used. The semi-structured survey guidelines we used in the interviews were derived from the validated instruments created for the validation of the *Agile Potential Analysis*. In addition to the instrument, another threat that was already mentioned indirectly is the higher degree of freedom in the interviews because we used open questions instead of closed ones (as in the first part).

The interview guidelines as well as the overall procedure of the simulation validation were peer-reviewed by two experts, one of whom had several years of experience with empirical studies. We also decided to conduct this study based on an example scenario similar to a walk-through (Wharton, et al., 1994). Furthermore, each interview was scheduled for half an hour and was also completed during that time.

External validity is treated easily because the duration was the same for all the participants because of the walk-through approach. Nonetheless, the fact that we had only four participants in this study limits its generalizability. Furthermore, as we only got the SMEs from the previous case studies as participants, the maturity and the organizational context were almost the same. On the other hand, this also makes the results of this study more generalizable for the specific context of SMEs. For larger organizations, additional interviews or studies would be necessary.

5.4 Summary

In this chapter, we discussed the empirical validation of the complete method developed and described within this thesis (cf. Table 23). This started with the general validation methodology, which was applied for both parts of the overall method, including the *Agile Potential Analysis* and the *Simulation of Process Improvements*. Both of these parts were then evaluated separately by detailing the evaluation methodology with its research design, describing the execution, and presenting the results at the end. Once these two parts had been completed, they were combined, compared, and interpreted. Together, they represent the complete method. Table 23 presents an overview of the validation of our hypotheses for both parts.

Table 23: Overview of the validation of the ACAPI approach

Hypothesis / Aspect	APA	Simulation
Goal Achievement (H1)	**+**	**n.a.**
Comprehension (H2)	**+**	**+**
Understandability	+	o / +
Reliability	+	+
Transparency	+	o
Acceptance (H3)	**+**	**+**
Usability	o / +	+
Ease of Use	o	o
Job Relevancy	+	o
Outcome Quality	o	+
Outcome	+	+

To summarize the results already discussed above, the empirical validation showed positive results overall for the complete method proposed in this thesis as well as its two individual parts. Goal achievement (H1) could not be aggregated on the overall level because it was not one of the hypotheses in our validation of the *Simulation of Process Improvements*.

Especially comprehension (H2) showed very positive results in both parts, with significantly positive results for the *Agile Potential Analysis* and results of a more qualitative nature for the *Simulation of Process Improvements*.

Furthermore, it was discussed positively in all cases and case studies. The resulting major benefit is that companies and their employees involved in an agile transformation have more trust and are thus more motivated because they comprehend how and why they are introducing the different agile elements. Nevertheless, the participants also mentioned some critical issues in the *Simulation of Process Improvements* part.

Finally, acceptance (H3) showed a similar picture as H2, only the other way around: Overall, it was seen as positive in both parts, but the detailed aspects of acceptance in the validation of the *Agile Potential Analysis* revealed some neutral results and even a very small number of negative results.

As the results were not positive for all the (sub-)aspects of the three hypotheses, we were also able to gather important feedback and information for improvements. The most important improvements that we were able to gather from of the empirical validation will be part of the future work described in the next chapter.

6 Conclusion and Future Work

"Genius begins great works,
labor alone finishes them."
Joseph Joubert

This chapter concludes the thesis by presenting a wrap-up of the different results. The summary of the thesis (Section 6.1) recapitulates the different contributions of this work that were initially sketched in the Introduction (Section 1.5.3). An outlook on possible future work and open questions identified during this work concludes the thesis (Section 6.2).

6.1 Results and Contributions

At the beginning of this thesis, we started by motivating software process improvement using single agile practices. From the practitioners' perspective, this is needed because in most cases, the current process does not address the improvement goals (PP1) and the selection of agile elements is done in an ad-hoc manner (PP2). Furthermore, from the scientific viewpoint, the lack of an evolutionary agile transition approach (SP1) and the lack of knowledge of the contribution of agile practices to constraints (SP2) were to be addressed. Having derived detailed requirements from the state of the practice, the state of the art, identified by means of literature studies, was compared with these requirements. Based on this preparatory work, the overall approach of this thesis was developed, which will be briefly summarized along with its major contributions in this section (cf. Section 1.5.3, Figure 43):

- The *Agile Practice Repository* (C1) with its schema and the *Agile Practice Impact Model* (APIM) describe individual agile practices in detail and present their relationship to improvement goals.

- The *simulation model* (C2) is built upon the APIM and extended by additional information obtained from company-specific experts.

- The *Agile Potential Analysis* (C3) is a goal-oriented and context-specific approach for selecting appropriate agile practices to support an evolutionary agile transition. It defines a step-by-step approach around the underlying repository and model.

- The *Simulation of Process Improvements* (C4) is an extension of the *Agile Potential Analysis* intended to enable data-based analysis of the combined impact of the selected improvement actions.

- To support the usage of this method, parts were (semi-)automated with *tool support* (C5), especially the *Agile Potential Analysis*.

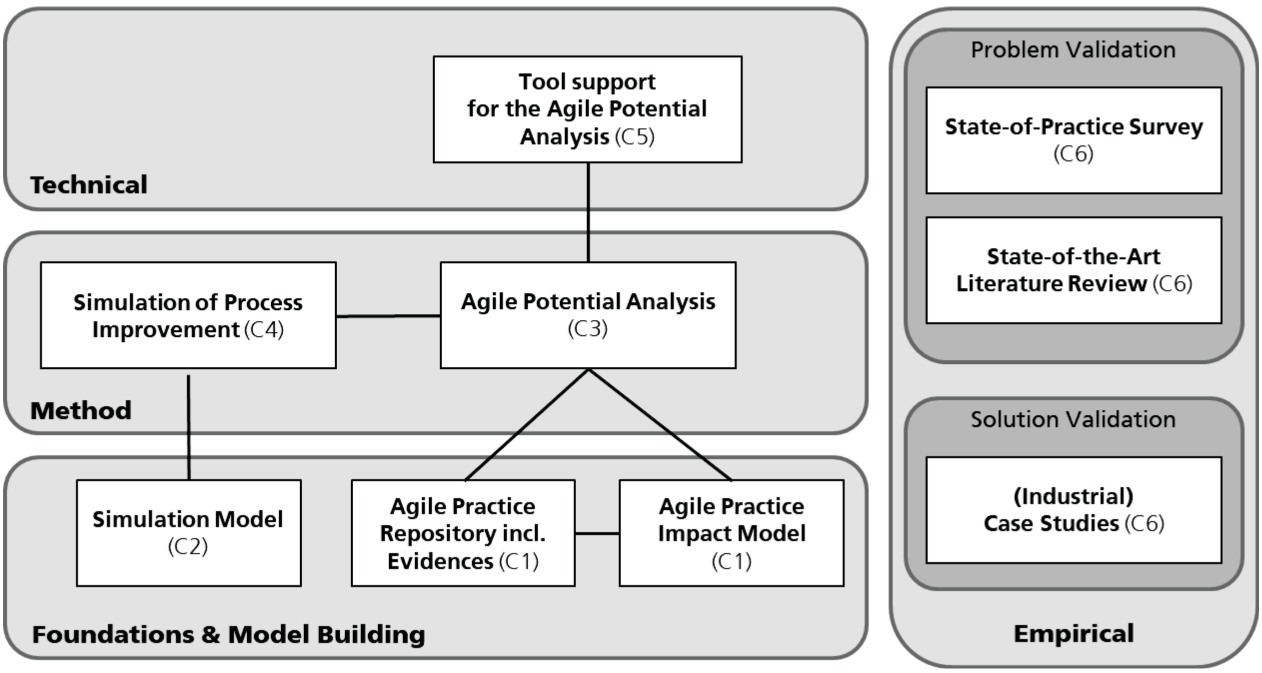


Figure 43: Overview of research contributions

Besides the more methodological contributions mentioned so far, the *problem* and *solution validation* (both C6) conducted here also form a research contribution. Six different industrial case studies were conducted for the empirical validation of the *Agile Potential Analysis*, supplemented by one case study for the *Simulation of Process Improvements*.

The outcome is an accepted approach that supports the initially stated problems: The empirical validation showed that using the ACPI approach helps to better address improvement goals (PP1). Furthermore, by using the approach, agile practices are no longer selected ad-hoc (PP2) but rather more systematically. From a scientific point of view, the approach supports an evolutionary agile transition approach (SP1). In addition, especially the underlying foundations and instances address the need for increasing knowledge about the contribution of agile practices to different constraints (SP2). Overall, the industrial case studies we conducted as well as the practical needs we identified (Diebold, et al., 2016) have shown the impact of this work.

6.2 Future Work and Open Questions

During the work on this thesis, future work aspects and possible open questions arose that could not be considered in the context of this thesis for different reasons. The following list includes all of them, clustered according to different topics. The first cluster that could be addressed by future work deals with the Agile Practice Repository:

- **Describing all practices of the repository in the schema** would be helpful for different use cases of the repository. So far we have only described the most commonly used practices in the complete schema, while the others only have the attributes needed for the *Agile Potential Analysis*, e.g. names, synonyms, and variation parameters.

- **Extending the repository with new practices** that might emerge from research in agile development or DevOps might lead to the creation of new suitable extensions. One good example would be an incremental safety certification to combine agility with (functional) safety development.

- **Enriching the existing APIM data** with more expert data than what we were able to collect at different events to date would be useful. Furthermore, it would also be possible to extend this data collection approach not only with regard to the core connection of the APIM but also for the surrounding connections, e.g., the context exclusions for the agile practices.

Strongly connected with the repository is the *Agile Potential Analysis*, which builds upon the APIM:

- **Improving the execution of the elicitation workshops** based on the lessons learned, especially those from the VSF Experts case study, could be very helpful:

    - **Predefining questions to elicit context**, e.g., from (Kalus & Kuhrmann, 2013) or (Theobald, 2016), could support discussion and elicitation. Having such guidelines or sets of questions would facilitate the procedure of eliciting contextual information.

    - **Naming as many things as possible in the customer's language** would make all the steps of the *Agile Potential Analysis* and *Simulation of Process Improvements* much easier because customers sometimes do not know the terms we are using in our models or approach, such as the difference between a goal and a characteristic. Thus, finding the right level to talk to each other is one of the toughest things to do in every meeting or workshop.

   - **Improving the presentation or explanation of the agile practices** would help all participants to have the same understanding. This is only necessary if the workshop is about more than just the elicitation. One example could be the usage or distribution of our pocket guides containing a one-slide description of each practice.

   - **Separating the elicitation of improvement goals and context** would make it easier for all the participants. Furthermore, if they are separated in two meetings (which is not mandatory), the first step of the goal elicitation could be performed completely first, yielding better input for the context elicitation.

- **Involving more different stakeholders in *Agile Potential Analysis*** would make the results more reliable. Independent of the three role categories that we defined in our validation(s), we found that different viewpoints especially at the beginning of the *Agile Potential Analysis* increases the reliability of the results.

- **Including more predefined connections in the *Agile Potential Analysis*** would help to include the experience of more experts, to automate more parts, and to reduce the time needed for conducting the analysis. Similar to some context characteristics that we integrated from the research project ProKoB[15], e.g., team size, or from the ASPICE-Mapping (Diebold, et al., 2017), more influences of characteristic could be integrated and thus contribute to a higher degree of automation and time reduction.

- **Improving the tool support for the *Agile Potential Analysis***, which is currently more a prototype than real tool support, would be another aspect. Due to our experiences with the visual vs. tabular representation, we would drop the EA add-on development

[15] www.prokob.info

and focus on the MS Excel tooling. Here, the goal should be to automate as many things as possible and finish the MS Excel add-on we started developing using VBA.

- **Elaborating further potential usage scenarios for the *Agile Potential Analysis*** would be useful, as stated by one of the case study participants. For example, in the VSF Experts case, the analysis was also used to question the correct application of the agile practices already in use.

In addition to the *Agile Potential Analysis* as the first part of the overall method, the *Simulation of Process Improvements* as well as its model also give rise to future work aspects:

- **Improving the visualization of the simulation results** could reduce or eliminate the feeling of high complexity and information overload. One example given by a study participant would be to increase interactive presentation of the results step by step or to make information visible only when it is selected.
- **Elaborating new usage scenarios of the *Simulation of Process Improvements*** (results) might motivate more people to use this part of the overall approach. The following examples came from the interview participants:
    - **Mitigating risks**, by knowing the upper and lower bound of the impacts of a practice and comparing this with your individual situation of using this practice.
    - **Promoting and distributing the overall approach** within an organization by having the visualizations of the results that are perceived to be very helpful.
    - **Combining simulation with Excel tooling for the *Agile Potential Analysis*** to foster interactivity. This can

be done very easily, especially with the parts of the *Simulation of Process Improvements* that are already performed in Excel: by doing so, we could also directly address the aspect of more interactivity.

Besides the methodological parts, there is also some potential future work for the empirical validations :

- **Improving the data collection instrument** (questionnaire for the *Agile Potential Analysis* and interview guidelines for the *Simulation of Process Improvements*) of the empirical validations. One easy and possible improvement would be the limitation of the questionnaire for specific roles, so that the process users would only need to rate output-related aspects and not everything.

- **Extending the validation of the simulation** by either increasing the number of interview participants in our current study or conducting more studies. In the latter case, it would be especially important to cover more companies with different characteristics, such as larger companies, or to cover more regulated domains, e.g. embedded domains.

Besides these concrete future aspects, which are connected directly with the topics of this thesis, there are some other interesting aspects that emerged during this work but were beyond its scope:

- **Supporting the Agile Transition** as the next step after performing the method presented in this thesis. As this is the next step in implementing "agility" in a company, it might be connected with the topic of this thesis: If the transition is conducted in an agile way (meaning incrementally and iteratively), the output of the method proposed in this thesis could be used at the beginning and during the transition as an agile process improvement.

- **Identifying and elaborating on the connection with the topic of interfaces from agile development in non-agile environments**. Besides transition support, which is more important for an individual team or a complete project, the general agile transition from an agile team via a project or program to a complete agile organization is important. A company starting such a journey needs to face different interfaces during the various stages (Theobald & Diebold, 2018). Since this thesis only supports parts of this, the connection and extension to cover agile practices at all organizational levels could be interesting.

These open questions and future work aspects conclude this thesis. We are looking forward to further applying our ACAPI approach in practice with different practitioners as well as to conducting further research on the topics of this thesis, including agile cherry-picking, agile transition, or interfaces between agile and non-agile development.

References

International Organization for Standardization, 2011. *ISO/IEC 15504 - Information technology - Process assessment,* s.l.: International Organization for Standardization.

Abrahamsson, P., Oza, N. & Siponen, M. T., 2010. Agile Software Development Methods: A Comparative Review. In: *Agile software development.* Berlin [u.a.]: Springer, p. 31–59.

Abrahamsson, P., Warsta, J., Siponen, M. T. & Ronkainen, J., 2003. New directions on agile methods: a comparative analysis. In: Los Alamitos, Calif.: IEEE Computer Society, p. 244–254.

Anderson, D. J., 2010. *Kanban.* Sequim, Washington: Blue Hole Press.

Arisholm, E., Gallis, H., Dyba, T. & Sjoberg, D. I., 2007. Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise. *IEEE Trans. Software Eng. (IEEE Transactions on Software Engineering)*, 33(2), p. 65–86.

Basili, V. R., Caldiera, G. & Rombach, H. D., 2008. Experience Factory. In: *Encyclopedia of software engineering.* 2nd ed. ed. [New York]: Wiley, p. 310.

Basili, V. R. & Rombach, H. D., n.d. Tailoring the software process to project goals and environments. In: *ICSE '87 Proceedings of the 9th international conference on Software Engineering.* Monterey, California, USA : ACM, p. 345–357.

Basili, V. et al., 2014. *Aligning Organizations Through Measurement - The GQM+Strategies Approach,* Heidelberg, Germany: Springer International Publishing.

Beck, K. & Andres, C., 2007. *Extreme programming explained.* 2. ed., 6. printing ed. Boston: Addison-Wesley.

Beck, K. et al., 2001. *Manifesto for Agile Software Development.* [Online] Available at: http://agilemanifesto.org/ [Accessed 8 8 2018].

Boehm, B. & Turner, R., 2003. *Balancing Agility and Discipline: A Guide for the Perplexed,* Boston, MA, USA: Addison-Wesley Longman Publishing.

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

Brown, A. W., Ambler, S. & Royce, W., 2013. Agility at scale: Economic governance, measured improvement, and disciplined delivery. In: Piscataway, NJ: IEEE, p. 873–881.

Chebyshev, P., 1867. On Mean Values. *Journal de Mathématiques Pures et Appliquées,* 2(12), pp. 177-184.

Clarke, P. & O'Connor, R. V., 2012. The situational factors that affect the software development process: Towards a comprehensive reference framework. *Information and Software Technology*, 54(5), p. 433–447.

Dantzig, G. B. & Thapa, M. N., 1997. *Linear Programming.* New York, NY: George B. Dantzig and Mukund N. Thapa.

Deming, W. E., 1982. *Out of the Crisis,* s.l.: Massachusetts Inst Technology.

Diebold, J., Diebold, P. & Vetter, A., 2018. *Agile meets Assessments: Case Study on how to do Agile Process Improvement in a Very Small Enterprise.* Heidelberg, Germany, Springer.

Diebold, P., 2014. *Initial Data Triangulation of Agile Practices Usage: Comparing Mapping Study and Survey Results.* s.l., Springer, Lecture Notes in Computer Science , pp. 269-272.

Diebold, P. & Dahlem, M., 2014. Agile practices in practice. In: New York, New York: Association for Computing Machinery, p. 1–10.

Diebold, P., Daun, M., Heuer, A. & Jedlitschka, A., 2016. Evaluation of the SPES XT Modeling Framework. In: *Advanced Model-Based Engineering of Embedded Systems.* Cham: Springer International Publishing;Imprint;Springer, p. 263–271.

Diebold, P., Galster, M., Rainer, A. & Licorish, S. A., 2017. Interactive Posters: An Alternative to Collect Practitioners' Experience. In: New York, NY: ACM, p. 230–235.

Diebold, P., Küpper, S. & Zehler, T., 2015. *Nachhaltige Agile Transition: Symbiose von technischer und kultureller Agilität.* Bonn, Germany, Gesellschaft für Informatik e.V., p. 121–126.

Diebold, P. & Mayer, U., 2017. *On the Usage and Benefits of Agile Methods & Practices – On the Usage and Benefits.* Heidelberg, Germany, Springer International Publishing.

Diebold, P., Mendez Fernandez, D., Smite, D. & Wagner, S., 2017. *ImpAct-2nd International Workshop on the Impact of Agile Practices.* s.l., s.n.

Diebold, P., Mendez Fernandez, D., Wagner, S. & Smite, D., 2017. *Results of the 2 nd International Workshop on the Impact of Agile Practices (ImpAct 2017).* s.l., ACM, p. 24.

Diebold, P., Ostberg, J.-P., Wagner, S. & Zendler, U., 2015. *What Do Practitioners Vary in Using Scrum?.* s.l., Springer - Lecture Notes in Business Information Processing , pp. 40-51.

Diebold, P. & Schmitt, A., 2016. *Why Do We Do Software Process Improvement?.* Heidelberg, Germany, Springer, p. 360–367.

Diebold, P., Schmitt, A. & Theobald, S., 2017. Mit Kontext- & Zielorientierung die Agile Transition erfolgreich gestalten. *Objekt spektrum. Online-Ausgabe*, Volume Agile.

Diebold, P., Schmitt, A. & Zehler, T., 2016. Agile Potenzialanalyse: Der Weg zum passenden Grad an Agilität. *Objekt spektrum. Online-Ausgabe*, Issue Sonderheft Agilität, p. 4.

Diebold, P., Smite, D. & Mendez Fernandez, D., 2015. Results of the 1st international workshop on impact of agile practices (ImpAct 2015). *ACM SIGSOFT Software Engineering Notes,* 40(6), pp. 20-23.

Diebold, P. & Theobald, S., 2018. How is agile development currently being used in regulated embedded domains?. *Journal of Software: Evolution and Process*, 26(5), p. e1935.

Diebold, P., Theobald, S., Schmitt, A. & Schmidt, C., 2017. Weg vom unvollständigen Scrum! Hin zum vollständigeren „Scrum++". *1617-5468*.

Diebold, P., Theobald, S., Wahl, J. & Rausch, Y., 2018. *An Agile transition starting with user stories, DoD & DoR.* s.l., s.n., p. 147–156.

Diebold, P. & Zehler, T., 2015. *The Agile Practices Impact Model: idea, concept, and application scenario.* New York, USA, ACM, p. 92–96.

Diebold, P. & Zehler, T., 2016. The Right Degree of Agility in Rich Processes. In: *Managing Software Process Evolution.* Cham: Springer International Publishing, p. 15–37.

Diebold, P., Zehler, T. & Richter, D., 2017. How do agile practices support automotive SPICE compliance?. In: New York, NY: ACM, p. 80–84.

Diebold, P. et al., n.d. Prozessverbesserung durch Fragmentierte Anwendung von Scrum & Co.. In: *Martin Engstler et al. (Hrsg.): Projektmanagement und Vorgehensmodelle 2016.* s.l.:s.n., p. 135–143.

Doubilet, P. et al., 1985. Probabilistic Sensitivity Analysis Using Monte Carlo Simulation. *Med Decis Making*, 5(2).

Dybå, T., Sjøberg, D. I. & Cruzes, D. S., 2012. What works for whom, where, when, and why?. In: New York, NY: ACM, p. 19.

Elorantaa, V.-P., Koskimies, K. & Mikkonen, T., 2016. Exploring ScrumBut—An empirical study of Scrum anti-patterns. *Information and Software Technology,* Volume 74, pp. 194-203.

Esfahani, H. C., 2015. *Transitioning to Agile: A Framework for Pre-adoption Analysis using Empirical Knowledge and Strategic Modeling,* s.l.: s.n.

Esfahani, H. C., Yu, E. & Annosi, M. C., 2010. Capitalizing on Empirical Evidence during Agile Adoption. In: Piscataway, NJ: IEEE, p. 21–24.

Fitzpatrick, S. & Scott, A., 1986. Quick Simultaneous Confidence Intervals for Multinomial Proportions. *Journal of the American Statistical Association ,* 82(399), pp. 875-878.

Fontana, R. M. et al., 2014. Processes versus people: How should agile software development maturity be defined?. *Journal of Systems and Software*, Volume 97, p. 140–155.

Garzás, J. & Paulk, M. C., 2013. A case study of software process improvement with cmmi-dev and scrum in spanish companies. *Journal of Software: Evolution and Process*, 25(12), p. 1325–1333.

George, S. & Weimerskirch, A., 1998. *Total Quality Management: Strategies and Techniques Proven at Today's Most Successful Companies.* 2nd ed. s.l.:Wiley.

Glaz, J. & Sison, C. P., 1999. Simultaneous confidence intervals for multinomial proportions. *Journal of Statistical Planning and Inference*, 82(1-2), p. 251–262.

Goodman, L. A., 1965. On Simultaneous Confidence Intervals for Multinomial Proportions. *Technometrics,* 7(2), pp. 247-254.

Grover, V., Cheon, M. J. & Teng, J. T., 1996. The Effect of Service Quality and Partnership on the Outsourcing of Information Systems Functions. *Journal of Management Information Systems*, 12(4), p. 89–116.

Guzmán Rehbein, L. K., n.d. *Empirically-based Method for Performing Qualitative Synthesis in Software Engineering,* TU: Kaiserslautern;Fraunhofer IESE;Fraunhofer IRB-Verlag.

Guzman, L., Vollmer, A. M., Ciolkowski, M. & Gillmann, M., 2017. Formative Evaluation of a Tool for Managing Software Quality. In: Piscataway, NJ: IEEE, p. 297–306.

Harry, M. J., 1988. *the Nature of Six Sigma Quality.* s.l.:Motorola University Press.

Haugen, N. C., 2006. An Empirical Study of Using Planning Poker for User Story Estimation. In: Los Alamitos, Calif.: IEEE Computer Society, p. 23–34.

Hodgetts, P., 2004. Refactoring the Development Process: Experiences with the Incremental Adoption of Agile Practices. In: [Place of publication not identified]: IEEE Computer Society, p. 106–113.

Houston, A. & Dockstader, S., 1988. *A Total Quality Management Process Improvement Model,* San Diego, CA, USA: Navy Personnel Research and Development Center.

Hulkko, H. & Abrahamsson, P., 2008. A multiple case study on the impact of pair programming on product quality. In: New York, NY: ACM, p. 495–504.

IEEE Computer Society, 1990. *IEEE Standard Glossary of Software Engineering Terminology,* Piscataway, NJ, USA: s.n.

IEEE Computer Society, 2014. *Guide to the Software Engineering Body of Knowledge Version 3.0 (SWEBOK Guide V3.0),* Wahsington, USA: IEEE Computer Society.

International Electrotechnical Commission (IEC), 2010. *IEC 61508 - Functional Sfatey of Electrical/Electronic/Programmable Electronic Safety-related Systems,* Geneva, Switzerland: International Electrotechnical Commission.

International Organization for Standardization, 2011. *ISO 26262: Road vehicles - Functional safety,* Geneva, Switzerland: International Organization for Standardization.

International Organization for Standardization, 2011. *ISO/IEC 25010:2011 - Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models,* Geneva, Switzerland: International Organization for Standardization.

Ivarsson, M. & Gorschek, T., 2011. A method for evaluating rigor and industrial relevance of technology evaluations. *Empirical Software Engineering*, 16(3), p. 365–395.

Ivarsson, M. & Gorschek, T., 2011. A method for evaluating rigor and industrial relevance of technology evaluations. *Empirical Software Engineering,* 16(3), pp. 365-395.

Jedlitschka, A. & Pfahl, D., 2005. Reporting guidelines for controlled experiments in software engineering. In: Los Alamitos, Calif.: IEEE Computer Society, p. 92–101.

Kalus, G. & Kuhrmann, M., 2013. Criteria for software process tailoring: a systematic review. In: New York, NY: ACM, p. 171.

Kirch, W., 2008. Level of Measurement. In: *Encyclopedia of public health.* New York, NY: Springer, p. 851–852.

Kitchenham, B., 2004. *Procedures for Performing Systematic Reviews,* s.l.: Keele University.

KItchenham, B. & Charter, S., 2007. *Guidelines for performing Systematic Literature Reviews in Software Engineering.* UK: s.n.

Kläs, M., 2011. *CoBRiX-Manual,* Kaiserslautern, Germany: Fraunhofer IESE.

Komus, A. & Kuberg, M., 2016. *Status Quo Agile,* Koblenz, Germany: s.n.

Kruchten, P., 2003. *The Rational Unified Process: An Introduction.* 3rd ed. s.l.:Addison-Wesley Professional.

Kruchten, P., 2013. Contextualizing agile software development. *Journal of Software: Evolution and Process*, 25(4), p. 351–361.

Kugler Maag CIE, 2015. *Agile in Automotive – State of Practice 2015,* Kornwestheim, Germany: Kugler Maag CIE.

Kuhrmann, M., Diebold, P. & Münch, J., 2016. Software process improvement: a systematic mapping study on the state of the art. *PeerJ Computer Science*, 2(1), p. e62.

Kuhrmann, M. et al., 2018. Hybrid Software Development Approaches in Practice: A European Perspective. *IEEE Software*, p. 1.

Kuhrmann, M. et al., 2015. Software process improvement: where is the evidence?: initial findings from a systematic mapping study. In: New York, NY: Association for Computing Machinery, Inc, p. 107–116.

Lee, Y. W. & Strong, D. M., 2003. Knowing-Why About Data Processes and Data Quality. *Journal of Management Information Systems*, 20(3), p. 13–39.

Madeyski, L., 2006. The Impact of Pair Programming and Test-Driven Development on Package Dependencies in Object-Oriented Design — An Experiment. In: *Product-focused software process improvement.* Berlin [u.a.]: Springer, p. 278–289.

Masaaki, I., 1986. *Kaizen: The Key to Japan's Competitive Success.* s.l.:McGraw-Hill Education.

May, W. L. & Johnson, W. D., 1997. A SAS® macro for constructing simultaneous confidence intervals for multinomial proportions. *Computer Methods and Programs in Biomedicine*, 53(3), p. 153–162.

Mc Hugh, M. et al., 2013. An agile V-model for medical device software development to overcome the challenges with plan-driven software development lifecycles. In: Piscataway, NJ: IEEE, p. 12–19.

McDowell, C., Werner, L., Bullock, H. E. & Fernald, J., 2003. The impact of pair programming on student performance, perception and persistence. In: Los Alamitos, Calif.: IEEE Computer Society, p. 602–607.

McHugh, M., McCaffery, F. & Casey, V., 2014. Adopting agile practices when developing software for use in the medical domain. *Journal of Software: Evolution and Process*, 26(5), p. 504–512.

McKinney, V., Yoon, K. & Zahedi, F. “., 2002. The Measurement of Web-Customer Satisfaction: An Expectation and Disconfirmation Approach. *Information Systems Research*, 13(3), p. 296–315.

Münch, J., Armbrust, O., Kowalczyk, M. & Soto, M., 2012. *Software Process Definition and Management.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Myklebust, T., Stålhane, T. & Lyngby, N., op. 2015. Application of an agile development processes for EN50128/Railway conformant software. In: *Safety and reliability of complex engineered systems.* Boca Raton; London: CRC Press;Taylor & Francis, p. 4037–4044.

Nash, S., ed., 1990. *History of scientific and numeric computation.* Reading, Ma: Addison-Wesley.

Nash, S. G. ed., 1990. *A history of scientific computing.* Reading, Mass.: Addison-Wesley Pub. Co.

Nering, E. D. & Tucker, A. W., 1993. *Linear programs and related problems.* Boston: Academic Press.

Niazi, M. et al., 2016. Toward successful project management in global software development. *International Journal of Project Management*, 34(8), p. 1553–1567.

O'hEocha, C. & Conboy, K., 2010. The Role of the User Story Agile Practice in Innovation. In: *Lean Enterprise Software and Systems.* Online-ausg ed. Berlin, Heidelberg: Springer Berlin Heidelberg, p. 20–30.

Object Management Group (OMG), 2008. *Software & Systems Process Engineering Meta-Model Specification.* Needham, MA 02494 USA: Object Management Group (OMG).

Office of the Assistant Secretary of Defense, 1965. *A Guide to Zero Defects. Quality and Reliability Assurance Handbook 4155.12-H.* s.l.:s.n.

Palisade, n.d. *What is Monte Carlo Simulation?.* [Online] Available at: http://www.palisade.com/risk/monte_carlo_simulation.asp [Accessed 8 8 2018].

Pannell, D., 1997. Sensitivity analysis of normative economic models: theoretical framework and practical strategies. *Agricultural Economics*, 16(2), p. 139–152.

Paternoster, N. et al., 2014. Software development in startup companies: A systematic mapping study. *nformation and Software Technology,* 56(10), pp. 1200-1218.

Petersen, K., Feldt, R., Mujtaba, S. & Mattsson, M., 2008. Systematic Mapping Studies in Software Engineering. In: *Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering.* s.l.:s.n., p. 68–77.

Petersen, K. & Wohlin, C., 2009. Context in industrial software engineering research. In: [Piscataway, N.J.]: IEEE, p. 401–404.

Petersen, K. & Wohlin, C., 2010. Software process improvement through the lean measurement (spi-leam) method. *Journal of Systems and Software*, 83(7), p. 1275–1287.

Project Management Institute, 2017. *PMBOK Guide - A Guide to the Project Management Body of Knowledge.* 6th ed. s.l.:Project Managemenr Institute (PMI).

Quesenberry, C. P. & Hurst, D. C., 1964. Large Sample Simultaneous Confidence Intervals for Multinomial Proportions. *Technometrics,* 6(2), pp. 191-195.

Rodriguez, P., Partanen, J., Kuvaja, P. & Oivo, M., 2014. Combining Lean Thinking and Agile Methods for Software Development: A Case Study of a Finnish Provider of Wireless Embedded Systems Detailed. In: Los Alamitos, California: Conference Publishing Services, IEEE Computer Society, p. 4770–4779.

Royce, W., 1970. *Managing the Development of Large Software Systems.* s.l., IEEE, pp. 1-9.

Runeson, P., ed., 2012. *Case study research in software engineering.* 1st ed. ed. Hoboken, N.J.: Wiley.

Runeson, P. & Höst, M., 2008. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), p. 131.

Runeson, P., Höst, M., Rainer, A. & Regnell, B., 2012. *Case Study Research in Software Engineering.* Hoboken, NJ, USA: John Wiley & Sons, Inc.

Runeson, P., Höst, M., Rainer, A. & Regnell, B., 2012. *Using Case Study Research.* s.l.:s.n.

Salinas, C. J. T., Escalona, M. J. & Mejías, M., 2012. A scrum-based approach to cmmi maturity level 2 in web development environments. In: New York: ACM, p. 282.

Salo, O. & Abrahamsson, P., 2005. Integrating agile software development and software process improvement: a longitudinal case study. In: Los Alamitos, Calif.: IEEE Computer Society, p. 187–196.

Salo, O. & Abrahamsson, P., 2007. An iterative improvement process for agile software development. *Software Process: Improvement and Practice*, 12(1), p. 81–100.

Saltelli, A., 2002. Sensitivity Analysis for Importance Assessment. *Risk Analysis*, 22(3), p. 579–590.

Saltelli, A. et al., 2008. *Global Sensitivity Analysis.* 1. Aufl. ed. s.l.: Wiley-Interscience.

Sato, D. et al., 2006. Experiences tracking agile projects: an empirical study. *Journal of the Brazilian Computer Society*, 12(3), p. 45–64.

Schwaber, K. & Sutherland, J., 2017. *The Scrum Guide,* s.l.: s.n.

Shaw, M., 2003. *Writing good software engineering research papers: minitutorial.* s.l.:s.n.

Sison, R. & Yang, T., 2007. Use of Agile Methods and Practices in the Philippines. In: Los Alamitos, Calif.: IEEE Computer Society, p. 462–469.

Snowden, D. J. & Boone, M. E., 2007. A Leader's Framework for Decision Making. *Harvard Business Review,* Issue November, pp. 69-76.

Software Engineering Institute, 2010. *CMMI for Development (CMMI-DEV),* s.l.: Carnegie Mellon University.

Sommerville, I., 2012. *Software Engineering.* 9th ed. s.l.:Person Studium.

Stålhanea, T., Myklebustb, T. & Hanssenb, G., 2012. The application of Safe Scrum to IEC 61508 certifiable software.

Stevens, S. S., 1946. On the Theory of Scales of Measurement. *Science (New York, N.Y.)*, 103(2684), p. 677–680.

Student, 1908. The Probable Error of a Mean. *Biometrika,* 6(1), pp. 1-25.

Theobald, S., 2016. *elicitation and Formalization of Goals and context to Support Software Process Improvement,* Kaiserslautern, Germany: s.n.

Theobald, S. & Diebold, P., 2018. Interface Problems of Agile in a Non-agile Environment. In: *Agile Processes in Software Engineering and Extreme Programming.* s.l.:Springer International Publishing, p. 123–130.

Trendowicz, A., 2013. *Software Cost Estimation, Benchmarking, and Risk Assessment.* Berlin; Heidelberg: Springer.

Trendowicz, A. & Münch, J., 2009. Factors Influencing Software Development Productivity – State-of-the-Art and Industrial Experiences. *Advances in Computers*, Volume 77, p. 185–241.

van Solingen, R., Basili, V., Caldiera, G. & Rombach, H. D., 2008. Goal Question Metric (GQM) Approach. In: *Encyclopedia of software engineering.* 2nd ed. ed. [New York]: Wiley, p. 758.

Vanderbei, R. J., 2008. *Linear programming.* 3. ed. ed. New York, NY: Springer.

VDA QMC Working Group 13, 2015. *Automotive SPICE - Process Reference Model and Process Assessment Model,* s.l.: s.n.

Venables, W. N., Smith, D. M. & the R Core Team, 2018. *An Introduction to R,* s.l.: s.n.

Venkatesh, V. & Bala, H., 2008. Technology Acceptance Model 3 and a Research Agenda on Interventions. *Decision Sciences*, 39(2), p. 273–315.

Verein Deutscher Ingenieure (VDI), 1996. *System zur Zukunftssicherung: Total Quality Management (TQM),* Düsseldorf, Germany: Verein Deutscher Ingenieure (VDI).

VersionOne, 2018. *12th Annual State of Agile Report,* s.l.: VersionOne.

Wharton, C., Rieman, J., Lewis, C. & Polson, P., 1994. The cognitive walkthrough method: a practitioner's guide. In: *Usability inspection methods.* s.l.:John Wiley & Sons, Inc, pp. 105-140.

Wieringa, R., Maiden, N., Mead, N. & Rolland, C., 2006. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements Engineering*, 11(1), p. 102–107.

Williams, A. & Rainer, A., 2017. Toward the use of blog articles as a source of evidence for software engineering research. In: New York, NY: ACM, p. 280–285.

Wohlin, C. et al., 2012. *Experimentation in Software Engineering.* Berlin: Springer.

Appendix A: Agile Practice Repository

In this section of the appendix, details on the agile practices repository, which underlies the APIM, are presented. This includes the list of the collected agile practices with more details, e.g., the sources / list of identification.

A.1 List of Agile Practices

Here the aggregated list of all the possible agile practices we found is presented that we integrated into the Agile Practice Repository. Having looked at the scientific literature as well as at different types of non-scientific literature, such as blogs (Williams & Rainer, 2017), not everyone might agree that all these practices are actually agile practices. Even though our overall approach focuses on integrating agility into existing processes, it does not really matter whether some of these practices might not be seen as agile practices by some people.

After checking possible synonyms or adaptations, we came up with Table 24, which also contains a mapping to the ISO12207 covering the development lifecycle:

Table 24: List of Agile Practices used in the APIM

Agile Practice (name)	ISO12207	Agile Practice Synonyms
"Just-in-time" requirements elaboration	7.1.2	
Acceptance Testing	7.2.4	
Active stakeholders	7.1.1	
Agile Games		
Apply Patterns Gently	7.1.3	
Apply the Right Artifact	7.1.1	
Architectural Runway		

Architecture Driven Development	7.1.1	Up front Architecture
Architecture Line	7.3.3	
Architecture Sprint		
Automated Builds	7.1.6	Automating software builds
Automated Tests	7.2.4	Test automation
Backlog	7.1.2	
Backlog grooming	7.1.2	Story time, Backlog refinement meeting
Behavior driven development	7.1.1	
Big Visible Charts	6.3.7	Information Radiators, informative work-space, Dashboard
Burn Chart	6.3.2	Burn Down Chart, Burn Up Chart, Progress chart
Clean Code		
Close or osmotic communication	6.2.4	
Co-located team	6.2.2	
Code and Test	7.1.5	Code & Tests
Code Inspections	7.2.4	
Coding Styleguides		
collaborative and cooperative ap-proach	6.2.4	
Collective Ownership	7.2.2	
Communities of Practice		
Component-based architectures	7.1.3	Used component-based architecture
Configuration Management	7.2.2	Version control, Source Control
Continuous Delivery		
Continuous Deployment	7.1.6	Incremental Deployment
Continuous Integration	7.1.6	Integrate often
Control changes to software	7.2.2	Embracing changing requirements
CRC Cards	7.1.3,	class-responsibility-collaboration cards
Create Several Models in Parallel	7.1.1	
Create Simple Content	7.2.1	
Cross-Functional Team		
customer focus group reviews	7.2.5	
Daily Meeting	6.3.2	Daily Scrum Meeting, Daily Standup, Scrum daily meetings, Scrum meeting, Stand up
Dedicated integration computer	6.2.2	

Definition of Done	7.2.4	"Done" Criteria, Definition of Done includes quality goals, Acceptance Criteria
Definition of Ready	7.1.2	
Depict Models Simply	7.2.1	
Design by Contract	7.1.3	
Developing by Feature	7.1.1	
Discard Temporary Models	7.2.1	
Display Models Publicly	7.2.1	
Document late	7.2.1	
Domain Driven Design	7.1.3,	
Domain Object Modeling	6.4.1	
Easy access to expert users	6.4.1	
Epic	6.4.1	
Exploratory Testing	7.2.4	
Extensive use of design patterns	7.1.1	
Extreme feedback devices	6.4.1	
Facilitation	6.2.4	
Feature Teams	6.3.1	
Feature-based planning	6.3.1	
Feature-driven Development	7.1.1	
Features in iteration customer-visible		
Fitness for business purpose is the essential criterion for acceptance	7.2.5	
Formalize Contract Models	7.2.1	
Frequent delivery	6.2.1	Frequent releases
Given-When-Then	7.2.4	
Holistic diversity strategy	6.2.4	
Impediments	6.3.2	Impediment Report, Impediment Log, Impediment Backlog
Incremental Design	6.4.3	Model in Small Increments, Emergent Design, Evolutionary Design, 118
Incremental Development	6.2.1	Incremental development cycles
Individual Class Ownership	7.2.2	Code ownership
Informal Design	7.1.3	
Inspection	7.2.4	
Integrated Dev/QA	7.1.5	
Integration Testing	7.2.4	
INVEST	7.1.2	

Issue Tracking / Bug Tracking	7.2.8	
Iterate to Another Artifact	7.1.1	
Iteration Reviews	7.2.4	Iteration Demo, Sprint Review Meeting, Sprint Review, Acceptance Meeting, Cycle-Time Analysis
Iterative development	6.2.1	Iterations, Short Iterations, Develop Software iteratively
Just rules	6.2.4	
Metaphor	6.4.1	Metaphor, System metaphor
Methodology-tuning technique	6.2.1	
Minimum Viable Product		
Mock Objects	7.1.5	
Model with Others	7.1.1	
Move People Around	6.3.6	
Negotiated Scope	6.3.1	Negotiated Scope contract
Niko-Niko calendar	6.3.7	
Off-Site Customer	6.2.4	
On-site customer	6.2.4	Daily customer/product manager involvement, customer always available, Real Customer Involvement
Open Workspace	6.2.2	Informative workspace
Organization-wide Process	6.2.1	
Organizational Training	6.2.4	
Pair Programming	7.1.5	
Paper prototyping	7.1.1	
Parallelism and flux	6.3.1	
Pay-per-Use	6.4.9	
Peer Reviews	7.2.4	
Personas	6.4.1	
Phasing and Pacing		
Planning Game	6.3.1	
Planning meeting	6.3.1	
Planning Poker	6.3.1	
Product Canvas		
Product Owner	6.4.1	Dedicated Product Owner
Product Vision board		
Project Chartering	6.3.1	Product Vision / Vision Statement
Project Monitoring & Control	6.3.2	Progress tracking/monitoring, Monitoring, Progress Reporting

Prototyping		
Prove it with Code	7.1.5	Spike solutions
Quick Design Session	7.1.3	
Reduce intermediate work products	7.1.1	
Refactoring	7.1.5	
Regular Builds	7.1.6	
Relative Estimation	6.3.1	Point Estimates, Points, nebulous units of time story points
Release planning		
Requirement Prioritization	6.4.2	Features in iteration customer valued
Requirements Management	6.4.2	Manage requirements
Retrospective	6.2.1	Heartbeat Retrospective, Reflection Workshop, Reflective Improvement
Revision and Review	6.2.1	
Root Cause Analysis	7.2.8	
Rules of Simplicity	7.1.5	
Scrum Master	6.2.4	
Scrum of Scrums	7.1.1	
Self-organizing team		
Shippable Increment	6.2.1	Increment, "Potentially shippable" features at the end of each iteration
Shrinking Teams	6.2.4	
Sign Up	6.3.1	
Simple Design	7.1.3,	
Single Code Base	7.2.2	
Single Sourcing Information	7.2.2	
Sitting Together / Common Workspace	6.2.2	Sit together, Common Work Space, Shared Office-space, Team Room
Slack	6.3.1	
Small Release Cycles	6.2.1	Small/short releases
Smoke Testing / Build Verification Test	7.2.4	
Social contract/working agreements	6.3.1	
Software Metrics / Code Metrics & Analysis	6.3.7	
Specification by Example		
Sprint Zero		
Staging	6.3.1	
Story Mapping	6.4.2	

Story splitting	6.4.2	
Sustainable Pace	6.3.1	40h week, Energetic/Energized work
Synchronous communication	6.3.6	Face to face communication
System Testing	7.2.4	
Tacit Knowledge	6.3.6	
Taskboard	6.3.1,	Scrum Board
Team	6.2.4	Scrum Team
Team Continuity	6.2.4	
Team documentation focuses on decisions rather than planning	7.2.1	
Team-Based Estimation	6.3.1	
Technical environment	6.2.2	
Templates and Standards	6.3.6	
Test Driven Design	7.1.3	Test-first Design
Test Driven Development	7.1.1	Test first Development, Test-First Programming, Code the unit test first
Three C's	7.1.2	3 C's
Three questions	6.3.2	
Timebox	6.2.1	Fixed Iteration Length, Fixed Sprints, Weekly Cycle, Quarterly Cycle, Sprint
Ubiquitous language	6.3.6	
Unit Testing	7.1.5	
Update only when it hurts	7.1.1	
Usability testing	7.2.5	
Usage Scenarios	6.4.1	
Use Cases	6.4.1	
Use the Simplest Tools	6.2.2	
User stories	7.1.2	Stories, requirements written as informal stories
User viewings	7.2.6	
User-Centered Focus		
Value Stream Mapping	6.3.1	
Velocity	6.3.2	Velocity chart, velocity tracking
Verify software quality	7.2.4	
Visually model software	7.1.1	
Wireframes		
Work-in-Progress Limit		

A.2 Schema for Describing Agile Practices

As described in the Agile Practice Repository section, we came up with a final schema of attributes to describe the agile practices such that they can be used and implemented in the best way. For doing this, two different schemas from Diebold and Zehler (2016) and from the research project ProKoB were used, compared, and aggregated into the final schema (Section 4.2.1).

The comparison of these two schemas is illustrated in Table 25:

Table 25: Comparison of agile practices schemas and marked attributes for the repository

Cat.	**Schema Attributes** (Diebold & Zehler, 2016)	**Schema Attributes** (ProKoB)
Description	**Name**	**Name**
	Synonym(s)	**Synonym(s) / Abbreviations / Translations**
	Purpose	Short description
	Description	**Long description**
	Role	
	Guidance	
Assessment		Addressed Goals
		Advantages
		Disadvantages
		Pitfalls
		Effort (for pilot, implementation, …)
		Risk factor
Context		**Contextual restriction**
	Precondition	**Precondition**
	Post-condition	**Post-condition**
	Variation parameter	**Variation parameter**
	Life-cycle process	**Process matrix** (“high-level life-cycle phases” and “PMI process groups” (Project Management Institute, 2017))
		Related (Other) Agile Practices
	Source	**Reference**

Besides this comparison of the two schemas, we also have some examples of agile practices described in these schemas. The first example is the agile practice Pair Programming described in the schema of (Diebold & Zehler, 2016) and shown in Table 26:

Table 26: Example description of Pair Programming in (Diebold & Zehler, 2016) schema

Schema Elements	**Description**
Name	Pair programming
Synonym	Pairing; Peer programming
Precondition	Available *Requirements* and *Architecture* / *Design*
Post-condition	*Code* with high quality and shared knowledge regarding the written code
Purpose	Pair programming is a dialog between two developers simultaneously programming and trying to implement better software (Beck & Andres, 2007) with additional knowledge sharing.
Description	In pair programming, two programmers develop software as a pair together on one workstation. The driver writes code while the other person, the observer, reviews each line of code as it is typed in.
Role	(Two) developers
Variation parameter	Experience of the two developers: expert-expert, expert-novice, novice-novice
Life-cycle process	SW Implementation Processes: Software Implementation Process, Software Construction Process
Tool	---
Source	(Beck & Andres, 2007)

The second example contains the practice User stories and is described in German because the research project in which it was developed used German as the project and dissemination language:

Table 27: Example German description of Pair Programming in ProKoB schema

Schema Elements	**Description**
Name	User stories
Synonyme / Abkürzungen / Übersetzungen	• Stories • Anforderungen
Beschreibung	User stories sind ein Konzept für die Beschreibung und das Management von Anforderungen. Jede User Story beschreibt eine Funktionalität einer Software aus der Sicht einer Rolle (z.B. Nutzer oder Käufer) und muss für diese Rolle einen klaren Wert liefern. User stories sollen durch den „Kunden“ geschrieben werden um am besten nach dem folgenden Muster von Mike Cohn aufgebaut zu werden. Als <Benutzerrolle> will ich <das Ziel>, so dass <Grund für das Ziel> User stories werden in der Sprache des Kunden, sie beschreiben das Was und nicht das Wie. Während der Entwicklung entstehen die Details im Dialog zwischen

	dem Kunden und dem Team. Jede User Story muss eine abschließbare Anforderung darstellen, sie darf nicht zu allgemein formuliert werden. Größere Themen können zu Beginn in Epics zusammengefasst werden und später dann in einzelne User stories heruntergebrochen werden. Grundlegend besteht die User Story aus drei Teilen: • „Karte" mit der Anforderung der Rolle. Beschreibung des Kerns der Anforderung in einem Satz nach oben vorgegebenem Muster. • Konversation. Die Verpflichtung des Teams an den Kunden sich im Detail über die Story zu unterhalten sobald es an die Umsetzung geht. Der Dialog passiert iterativ während der Entwicklung bis diese abgeschlossen ist. • Akzeptanzkriterien, Tests. Beinhalten Details zur Umsetzung und legt fest wann eine User Story vollständig umgesetzt ist. Siehe auch Baustein Definition of Done. User stories können nach dem INVEST Schema erstellt werden: • Independent, User stories sollen unabhängig voneinander sein. • Negotiable, User stories sollen verhandelbar sein. Stories können zum Beispiel durchaus später verkleinert und Inhalte in neue Stories ausgelagert werden. • Valuable, User stories sollen einen Wert für den Kunden haben. Nach der Umsetzung jeder einzelnen User Story muss sich für den Anwender ein klarer Nutzen ergeben. • Estimable, User stories sollen schätzbar sein. • Small, User stories sollen klein sein. Die Entwicklungsdauer sollte zwischen einem halben Tag und maximal 2 Wochen liegen. • Testable, User stories brauchen Akzeptanzkriterien zur Prüfung der vollständigen Umsetzung der User Story. Die User stories werden nach dem Geschäftswert priorisiert und damit die Reihenfolge der Umsetzung festgelegt.
Zielfokus	• **Kundeneinbeziehung**: Kundenmitsprache, Kundenzufriedenheit, Projekttransparenz • **Organisations-Demokratisierung**: Projekt-Demokratisierung, internes Wissensmanagement • **Qualität**: Dokumentation, Testbarkeit / Abnahmekriterien, User Experience
Vorteile	Der Kunde wird von Anfang an in den Entwicklungsprozess einbezogen. Er wird zu jeder Zeit im Entwicklungsprozess gehört und wirkt dadurch direkt an der Software mit. Dadurch wird die Kundenmitsprache verbessert, das Projekt für den Kunden transparenter und das Ergebnis wird besser zu den späteren Nutzern passen. Software wird nicht anhand evtl. schon veralteter Anforderungen umgesetzt. Das Ergebnis wird besser das Treffen was der Kunde auch benötigt. Das Projekt konzentriert sich auf den Geschäftswert der Software für den Kunden und steigert damit die Akzeptanz für das Ergebnis. User stories sind allgemein verständlich, haben die richtige Größe für die Planung, eignen sich für iterative Softwareentwicklung, ermutigen dazu Details zurückzustellen, unterstützen opportunistisches Design und bauen implizites Wissen auf. Abnahmekriterien werden schon bei der Ausarbeitung der User stories festgelegt, sie definieren die Details der Umsetzung, und die Software kann dadurch vollständig getestet werden.
Nachteile	

Lokalisierung in der Prozessmatrix	Anforderung x Planung, Umsetzung, und Controlling Entwurf, Implementierung, Test x Planung
Vorbedingungen	Der Kunde muss dem Team für den direkten Dialog im Projekt zur Verfügung stehen. Rollen müssen bekannt sein (z.B. über den Baustein Rollenmodellierung).
Nachbedingungen	User stories sind dokumentiert.
Variations-parameter	Methoden zur Ermittlung von User stories: Interviews mit Usern, Fragebögen, Beobachtungen, Story-Workshops. Die Art der Dokumentation kann variieren, z.B. Post-It, Excel, Tools.
Kontext-eingrenzung	Kundenverfügbarkeit: wöchentlich
Fallstricke	Findet die aktive Kommunikation im Team nicht statt, und ist der Kunde nicht regelmäßig Verfügbar, dann ist die erfolgreiche Baustein Nutzung gefährdet. Bei der Anforderungserhebung kann es passieren, dass die die nichtfunktionalen Anforderungen vernachlässigt werden.
Risikobewertung	2 - Der wichtigste Faktor beim Arbeiten mit User stories ist die Konversation, das aktive Gespräch zwischen Kunden und Entwicklern. Bei der Kommunikation gibt es viele Faktoren, die diese beeinflussen (siehe. Vier-Ohren- Modell von Schulz von Thun)`, und damit viele Faktoren, an denen das Vorgehen scheitern kann. Wird der Baustein im schon fortgeschrittenen Projekt eingeführt kann dies schnell zum Chaos führen und erhöht das Risiko zusätzlich.
Undo-Schritte	Soll der Baustein während des Projektes wieder abgeschafft werden, muss hier in vielen Punkten wieder von vorne begonnen werden. Grundlegend kann der Baustein aber ohne sonstige Kosten oder Aktionen einfach wieder abgeschafft werden.
Aufwand für den ersten Durchstich	2 - Das Team muss das Prinzip verstehen und sich an die neue Herangehensweise gewöhnen. Der Baustein ist in sich aber nicht komplex und das Team wird die Änderung schnell annehmen können.
Aufwand für die Etablierung	2 - Siehe erster Durchstich.
Aufwand für den Regelbetrieb	3 - Der Aufwand wird bei den ersten Einsätzen nicht ganz gering sein, weil sich das Team erst mal an die neue Vorgehensweise gewöhnen muss. Der Mehrwert wird sich aber schnell einstellen, sodass die entstehenden Aufwände für den Betrieb an anderer Stelle wieder eingespart werden können.
Referenzen	(1) vgl. Scrum mit Users Stories von Ralf Wirdemann, 2. Auflage, Hanser Verlag. (2) vgl. User stories für die agile Software-Entwicklung mit Scrum, XP u.a., Mike Cohn, mitp Verlag (3) https://de.wikipedia.org/wiki/Vier-Seiten-Modell

More agile practices described in this schema in German can be found on the project website: www.prokob.info.

A.3 Evidences on Agile Practices

Table 28 presents the events where we performed the interactive poster approach (Diebold, et al., 2017) to collect impacts on agile practices:

Table 28: Events where we conducted the interactive poster approach

Name	Date	Contact	Participants		Country
Agile in Automotive 2016	2016-11-15	Philipp Diebold	170	Practitioner	Germany
PROFES 2016	2016-11-22	Philipp Diebold	150	Practitioner, Academic	International
OOP 2017	2017-01-31	Philipp Diebold	1500	Practitioner	Germany
Lean IT Management 2017	2017-03-08	Sven Theobald	100		Germany
AgileXchange 1-2017	2017-03-09	Sven Theobald	120	Practitioner	Germany
AgileLab Copenhagen	2017-03-22	Paolo Tell	50		Denmark
Q-Rapids Meeting	2017-05-11	Liliana Guzman	20	Practitioner, Academic	International
Agile in Automotive USA 2017	2017-05-16	Kevin Dibble		Practitioner	USA
CESI 2017	2017-05-22	Andreas Jedlitschka	10	Practitioner	International
XP 2017	2017-05-22	Philipp Diebold	280	Practitioner, Academic	International
ScrumDay 2017	2017-05-30	Philipp Diebold		Practitioners	Germany
EASE 2017	2017-06-15	Sherlock Licorish		Academics	International
AgileXchange 2-2017	2017-06-22	Philipp Diebold	60	Practitioners	Germany
SPA 2017	2017-06-26	Giovanni Asproni	50	Practitioner	UK
AgileAustria 2017	2017-06-28	Robert Herzig	250	Practitioner	Austria
Agile on the Beach 2017	2017-07-05	Belinda Waldock	400		UK
ICSSP 2017	2017-07-05	Philipp Diebold	35	Academics	
Agile Auckland 7-17	2017-07-26				New Zealand
ALE 2017	2017-08-28				Czech Republic
LASD 2017	2017-09-03	Adam Przybylek	20		Czech Republic
RE 2017	2017-09-04	Anne Hess			

Agile Brazil 2017	2017-09-13	Gustavo Pinto			Brazil
LeSS Conference 2017	2017-09-13	Philipp Diebold		Practitioner	United Kingdom
Lean, Agile & Scrum 2017	2017-09-14	Martin Kropp		Practitioner	
AgileXchange 3-2017	2017-09-21		70		Germany
Bitkom Forum Software 2017	2017-09-21				
Agile Summit Greece 2017	2017-09-22				Greece
Agile Bodensee 2017	2017-09-27	Philipp Diebold			Germany
PVM 2017	2017-10-05	Sven Theobald		Practitioners, Academics	Germany
Agile Tour London 2017	2017-10-20	David Gimelle			UK
ESEM 2017	2017-11-08	Liliana Guzman		Practitioners, Academics	International
Agile in Automotive 2017	2017-11-15	Markus Müller	150	Practitioners	Germany
GDCR Cologne 2017	2017-11-18	Steve Korzinetzki			Germany
Agile Automotive PEP 2017	2017-11-23	Philipp Diebold		Practitioner	Germany
PROFES 2017	2017-11-29	Philipp Diebold		Practitioners, Academics	
XP2018	2018-05-21	Philipp Diebold			

Due to tremendous amount of space needed to present the complete individual data set of the collected impacts using the interactive poster approach, they are attached as a digital appendix of this thesis: https://doi.org/10.5281/zenodo.1422800. The aggregated data is visualized at http://impact.iese.fhg.de/data.php.

In addition to the evidence collection from the posters, we started a workshop series with two instances around this topic and the data. Two "Impact" Workshops were performed, one at ICSSP2015 in Tallinn, Estonia (Diebold, et al., 2015), and the other at XP2017 in Cologne, Germany (Diebold, et al., 2017) (Diebold, et al., 2017).

Appendix B: Validation

B.1 Questionnaire for the Validation(s)

B.1.1 English Questionnaire for *Agile Potential Analysis*

Evaluation:
"Agile Potential Analysis"

Fraunhofer
IESE

Participant ID	
Participants role:	< Executer of the Potential Analysis, Process owner, Process user (e.g. RE, developer, tester, etc) >
Date:	

Dear participant,

In the context of the development of the Agile Potential Analysis, we are conducting a survey after each Application of the Agile Potential Analysis.

The goal of this survey is to understand the degree of comprehension and acceptance of the Agile Potential Analysis and its outcomes from the perspective of process owners as well as process users. Your feedback is important to better understand the usage of the Agile Potential Analysis method and to identify suggestions for improvements.

You have been selected to participate in this survey, because you were applying the methods or using its resulting outcomes. Therefore, we would like you to fill out the following questionnaire and give it back to Philipp Diebold (philipp.diebold@iese.fraunhofer.de) before day.month.year.

Your answers and your personal data will be treated confidentially. Furthermore, all survey results will be anonymized and only presented in an aggregated way.

The results of this survey will be published as part of my PhD Thesis "Agile Practice Experience Repository for Process Improvement" and in software engineering conferences or journals.

If you have any questions related to the method, its output, or this survey, do not hesitate to contact Philipp Diebold (philipp.diebold@iese.fraunhofer.de).

Thank you very much!

Philipp Diebold
Fraunhofer IESE, Process Engineering (PE)
Fraunhofer-Platz 1 | 67663 Kaiserslautern | Germany
Phone: +49 631 / 6800-2183 | Fax: +49 631 / 6800-9-2183
Mail: philipp.diebold@iese.fraunhofer.de

Evaluation: „Agile Potential Analysis"

Goal achievement

1. Using the Agile Potential Analysis helped me or my organization to …	Much better 5	Better 4	Same as before 3	Worse 2	Much worse 1	I don't know
… select appropriate improvement actions / items (i.e. agile practices)	☐	☐	☐	☐	☐	☐
… achieving our improvement goals	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… achieving ______	☐	☐	☐	☐	☐	☐
… in general, achieve our (software) process improvement goals	☐	☐	☐	☐	☐	☐
… achieve new improvement goals	☐	☐	☐	☐	☐	☐

2. Using the Agile Capability Analysis I can better improve …

3. I will recommend …

Fraunhofer IESE 2

Evaluation: „Agile Potential Analysis“

Understandability

	5 Very	4 Quite	3 Fairly	2 Slightly	1 Not at all	I don't know
4. How understandable do you consider the Agile Potential Analysis' <u>purpose</u>?	☐	☐	☐	☐	☐	☐
5. How understandable do you consider the Agile Potential Analysis' <u>process</u>?	☐	☐	☐	☐	☐	☐

6. Based on the results provided to you by the Agile Potential Analysis, do you expect information to be …	5 Extremely likely	4 Likely	3 Neutral	2 Unlikely	1 Extremely unlikely	I don't know
… clear in meaning	☐	☐	☐	☐	☐	☐
… easy to comprehend	☐	☐	☐	☐	☐	☐
… easy to read	☐	☐	☐	☐	☐	☐
… in general, <u>understandable</u> for making decisions regarding Agile Process Improvement	☐	☐	☐	☐	☐	☐

Reliability

7. Based on the results provided to you by the Agile Potential Analysis, do you expect information to be …	5 Extremely likely	4 Likely	3 Neutral	2 Unlikely	1 Extremely unlikely	I don't know
… trustworthy	☐	☐	☐	☐	☐	☐
… accurate	☐	☐	☐	☐	☐	☐
… credible	☐	☐	☐	☐	☐	☐
… in general, <u>reliable</u> for making decisions regarding Agile Process Improvement	☐	☐	☐	☐	☐	☐

Evaluation: „Agile Potential Analysis"

Transparency

To what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
8. The results are useful to our work.	☐	☐	☐	☐	☐	☐
9. The results are relevant to our work.	☐	☐	☐	☐	☐	☐
10. The results are appropriate for our work.	☐	☐	☐	☐	☐	☐
11. The results are applicable to our work.	☐	☐	☐	☐	☐	☐

Acceptance

Regarding the usability of the Agile Potential Analysis, to what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
12. Using the Agile Potential Analysis improves my performance in my job.	☐	☐	☐	☐	☐	☐
13. Using the Agile Potential Analysis in my job increases my productivity.	☐	☐	☐	☐	☐	☐
14. Using the Agile Potential Analysis enhances my effectiveness in my job.	☐	☐	☐	☐	☐	☐
15. I find the Agile Potential Analysis to be useful in my job.	☐	☐	☐	☐	☐	☐

Regarding the ease of use of the Agile Potential Analysis, to what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
16. My interaction with the Agile Potential Analysis is clear and understandable.	☐	☐	☐	☐	☐	☐
17. Interacting with the Agile Potential Analysis does not require a lot of my mental effort.	☐	☐	☐	☐	☐	☐
18. I find the Agile Potential Analysis to be easy to use.	☐	☐	☐	☐	☐	☐
19. I find it easy to get the Agile Potential Analysis to do what I want it to do.	☐	☐	☐	☐	☐	☐

Fraunhofer IESE 4

Evaluation: „Agile Potential Analysis"

Regarding the job relevance of the Agile Potential Analysis, to what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
20. In my job, usage of the Agile Potential Analysis is important.	☐	☐	☐	☐	☐	☐
21. In my job, usage of the Agile Potential Analysis is relevant.	☐	☐	☐	☐	☐	☐
22. The use of the Agile Potential Analysis is pertinent to my various job-related tasks.	☐	☐	☐	☐	☐	☐

Regarding the quality of the outcomes of the Agile Potential Analysis, to what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
23. The quality of the output I get from the Agile Potential Analysis is high.	☐	☐	☐	☐	☐	☐
24. I have no problem with the quality of the Agile Potential Analysis's output.	☐	☐	☐	☐	☐	☐
25. I rate the results from the Agile Potential Analysis to be excellent.	☐	☐	☐	☐	☐	☐

Regarding the outcomes of the Agile Potential Analysis, to what degree you agree or disagree with the following statements:	Strongly agree 5	Agree 4	Neither agree nor disagree 3	Disagree 2	Strongly disagree 1	I don't know
26. I have no difficulty telling others about the results of using the Agile Potential Analysis.	☐	☐	☐	☐	☐	☐
27. I believe I could communicate to others the consequences of using the Agile Potential Analysis.	☐	☐	☐	☐	☐	☐
28. The results of using the Agile Potential Analysis are apparent to me.	☐	☐	☐	☐	☐	☐
29. I would have difficulty explaining why using the Agile Potential Analysis may or may not be beneficial.	☐	☐	☐	☐	☐	☐

Other

30. Further comments or feedback:

B.1.2 German Questionnaire for *Agile Potential Analysis*

Evaluation: "Agile Potenzialanalyse"

Teilnehmer ID	
Teilnehmerrolle:	< Ausführer der Potenzialanalyse, Prozessverantwortlicher, Prozessnutzer (z.B. Entwickler) >
Datum:	

Lieber Teilnehmer,

im Rahmen der Entwicklung der Agilen Potenzialanalyse führen wir eine Umfrage zu deren Anwendung durch.

Das Ziel dieser Umfrage ist es, den Grad an Verständnis und Akzeptanz der Potenzialanalyse und ihrer Ergebnisse zu verstehen. Ihre Rückmeldung ist daher sehr wichtig, um die Benutzung der Agilen Potenzialanalyse zu verstehen und um Verbesserungsvorschläge zu identifizieren.

SIe wurden für die Teilnahme an dieser Umfrage ausgesucht, da Sie entweder die Methode angewendet haben oder die daraus resultierenden Ergebnisse eingesetzt haben. Aus diesem Grund würden wir Sie bitten, den vorliegenden Fragebogen auszufüllen und bis zum 02.02.2018 an Philipp Diebold (philipp.diebold@iese.fraunhofer.de) zurückzuschicken.

Ihre Antworten wie auch Ihre persönliche Daten werden vertraulich behandelt. Außerdem werden alle Umfrageergebnisse anonymisiert und nur in aggregierter Form präsentiert.

Die Ergebnisse dieser Umfrage werden im Rahmen meiner Doktorarbeit "Agile Practice Experience Repository for Process Improvement" und ggf. auch auf Software-Engineering-Konferenzen veröffentlicht.

Sollten Sie Fragen zu der Methode, dem Ergebnis der Methode oder der Umfrage haben, wenden Sie sich bitte an Philipp Diebold (philipp.diebold@iese.fraunhofer.de).

Vielen Dank!

Philipp Diebold
Fraunhofer IESE, Process Engineering (PE)
Fraunhofer-Platz 1 | 67663 Kaiserslautern
Tel: +49 631 / 6800-2183 | Fax: +49 631 / 6800-9-2183
E-Mail: philipp.diebold@iese.fraunhofer.de

Evaluation: „Agile Potentialanalyse"

Zielerreichung

1. Die Nutzung der Agilen Potenzialanalyse hilft mir oder meinem Unternehmen bei(m) …	deutlich besser 5	besser 4	genau wie vorher 3	schlechter 2	deutlich schlechter 1	Ich weiß es nicht
… der Auswahl passender Verbesserungspraktiken (d.h. Agiler Praktiken)	☐	☐	☐	☐	☐	☐
… Erreichen unserer Verbesserungsziele	☐	☐	☐	☐	☐	☐
… Verbessern der "Fehlender Projektüberblick"	☐	☐	☐	☐	☐	☐
… Erreichen von "Übersetzung der Wünsche in konkrete Aufgaben"	☐	☐	☐	☐	☐	☐
… Erreichen von "Kommunikation (innerhalb eines Teams und über dessen Grenzen hinweg)"	☐	☐	☐	☐	☐	☐
… Erreichen von "Wissenstransfer"	☐	☐	☐	☐	☐	☐
… Erreichen von "Verständnis beim Kunden"	☐	☐	☐	☐	☐	☐
… Erreichen von "Dokumentation"	☐	☐	☐	☐	☐	☐
… generellen Erreichen unsere (Software-) Prozessverbesserungsziele	☐	☐	☐	☐	☐	☐
… Erreichen neuer Verbesserungsziele	☐	☐	☐	☐	☐	☐

2. Durch die Nutzung der Agilen Potenzialanalyse kann ich besser …

3. Ich empfehle …

Fraunhofer IESE 2

Evaluation: „Agile Potentialanalyse"

Verständlichkeit

	ganz 5	ziemlich 4	ordentlich 3	etwas 2	gar nicht 1	Ich weiß es nicht
4. Wie verständlich ist für Sie der Zweck der Agilen Potenzialanalyse?	☐	☐	☐	☐	☐	☐
5. Wie verständlich ist für Sie der Prozess /Ablauf der Agilen Potenzialanalyse?	☐	☐	☐	☐	☐	☐

6. Basierend auf den Ergebnissen der Agilen Potenzialanalyse, betrachten Sie die Informationen als …	sehr wahr-scheinlich 5	wahr-scheinlich 4	neutral 3	unwahr-scheinlich 2	sehr unwahr-scheinlich 1	Ich weiß es nicht
… eindeutig	☐	☐	☐	☐	☐	☐
… leicht zu erfassen	☐	☐	☐	☐	☐	☐
… leicht zu deuten	☐	☐	☐	☐	☐	☐
… generell verständlich als Entscheidungsgrundlage für Agile Prozessverbesserung	☐	☐	☐	☐	☐	☐

Zuverlässigkeit

7. Basierend auf den Ergebnissen der Agilen Potenzialanalyse, betrachten Sie die Informationen als …	sehr wahr-scheinlich 5	wahr-scheinlich 4	neutral 3	unwahr-scheinlich 2	sehr unwahr-scheinlich 1	Ich weiß es nicht
… vertrauenswürdig	☐	☐	☐	☐	☐	☐
… exakt	☐	☐	☐	☐	☐	☐
… glaubwürdig	☐	☐	☐	☐	☐	☐
… generell zuverlässig als Entscheidungsgrundlage für Agile Prozessverbesserung	☐	☐	☐	☐	☐	☐

Evaluation: „Agile Potentialanalyse"

Transparenz

In welchem Maße stimmen Sie den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
8. Die Ergebnisse sind nützlich für unsere Arbeit.	☐	☐	☐	☐	☐	☐
9. Die Ergebnisse sind relevant für unsere Arbeit.	☐	☐	☐	☐	☐	☐
10. Die Ergebnisse sind passend für unsere Arbeit.	☐	☐	☐	☐	☐	☐
11. Die Ergebnisse sind anwendbar für unsere Arbeit.	☐	☐	☐	☐	☐	☐

Akzeptanz

In welchem Maße stimmen Sie in Bezug auf die Benutzbarkeit der Agilen Potenzialanalyse den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
12. Die Nutzung der Agilen Potenzialanalyse verbessert meine Leistung bei der Arbeit.	☐	☐	☐	☐	☐	☐
13. Die Nutzung der Agilen Potenzialanalyse bei meiner Arbeit verbessert meine Produktivität.	☐	☐	☐	☐	☐	☐
14. Die Nutzung der Agilen Potenzialanalyse verbessert die Effektivität meiner Arbeit.	☐	☐	☐	☐	☐	☐
15. Ich finde die Agile Potenzialanalyse nützlich für meine Arbeit.	☐	☐	☐	☐	☐	☐

In welchem Maße stimmen Sie in Bezug auf den Bedienkomfort der Agilen Potenzialanalyse den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
16. Meine Interkation mit der Agilen Potenzialanalyse ist klar und verständlich.	☐	☐	☐	☐	☐	☐
17. Der Umgang mit der Agilen Potentialanalyse bedarf keiner großen geistigen Anstrengung.	☐	☐	☐	☐	☐	☐
18. Ich empfinde die Agile Potenzialanalyse als bedienerfreundlich.	☐	☐	☐	☐	☐	☐
19. Ich empfinde es als einfach, das von der Agilen Potenzialanalyse zu bekommen, was ich von ihr erwarte.	☐	☐	☐	☐	☐	☐

Fraunhofer IESE 4

Evaluation: „Agile Potentialanalyse"

In welchem Maße stimmen Sie in Bezug auf die Relevanz der Agilen Potenzialanalyse für Ihre Arbeit den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
20. Bei meiner Arbeit ist die Nutzung der Agilen Potenzialanalyse wichtig.	☐	☐	☐	☐	☐	☐
21. Bei meiner Arbeit ist die Nutzung der Agilen Potenzialanalyse relevant.	☐	☐	☐	☐	☐	☐
22. Die Nutzung der Agilen Potenzialanalyse ist für meine verschiedenen Arbeitsaufgaben zweckdienlich.	☐	☐	☐	☐	☐	☐

In welchem Maße stimmen Sie in Bezug auf die Qualität der Ergebnisse der Agilen Potenzialanalyse den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
23. Die Qualität der Ergebnisse aus der Agilen Potenzialanalyse ist hoch.	☐	☐	☐	☐	☐	☐
24. Ich habe kein Problem mit der Qualität der Ergebnisse der Agilen Potenzialanalyse.	☐	☐	☐	☐	☐	☐
25. Ich bewerte die Ergebnisse der Agilen Potenzialanalyse als exzellent.	☐	☐	☐	☐	☐	☐

In welchem Maße stimmen Sie in Bezug auf die Ergebnisse der Agilen Potenzialanalyse den folgenden Aussagen zu oder widersprechen diesen:	stimme voll zu 5	stimme zu 4	weder noch 3	stimme nicht zu 2	stimme gar nicht zu 1	Ich weiß es nicht
26. Ich habe keine Probleme damit, anderen über die Ergebnisse der Nutzung der Agilen Potenzialanalyse zu erzählen.	☐	☐	☐	☐	☐	☐
27. Ich glaube, dass ich anderen die Folgen der Nutzung der Agilen Potenzialanalyse kommunizieren könnte.	☐	☐	☐	☐	☐	☐
28. Die Ergebnisse der Nutzung der Agilen Potenzialanalyse sind mir ersichtlich.	☐	☐	☐	☐	☐	☐
29. Ich hätte Schwierigkeiten zu erläutern, wieso die Nutzung der Agilen Potenzialanalyse nützlich bzw. nicht nützlich ist.	☐	☐	☐	☐	☐	☐

Sonstiges

30. Weitere Kommentare oder Feedback:

B.1.3 Interview Guidelines for *Simulation of Process Improvements*

Interview zum Simulations-AddOn der Agilen Potenzialanalyse

Einschätzungen im Vergleich zur Agilen Potenzialanalyse

- Wie schätzen Sie dieses AddOn im Vergleich zur Potenzialanalyse generell ein?
- Welchem Mehrwert sehen Sie in diesem AddOn im Vergleich zur Potenzialanalyse?
  -

Verständnis (=Comprehension)

- Wie schätzen Sie die Verständlichkeit des AddOn im Vergleich zur Potenzialanalyse ein?
  -
- Wie schätzen Sie die Transparenz des AddOn im Vergleich zur Potenzialanalyse ein?
  -
- Wie schätzen Sie die Zuverlässigkeit des AddOn im Vergleich zur Potenzialanalyse ein?
  -

Akzeptanz (=Acceptance)

- Wie schätzen Sie die Relevanz des AddOn im Vergleich zur Potenzialanalyse ein?
  - Wofür relevant? Bestimmte Entscheidungen? …?
- Wie schätzen Sie das Ergebnis des AddOn im Vergleich zur Potenzialanalyse ein?
  -
- Wie schätzen Sie die Qualität des Ergebnisses des AddOn im Vergleich zur Potenzialanalyse ein?
  -
- Wie schätzen Sie die Benutzbarkeit des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Bedienkomfort des AddOn im Vergleich zur Potenzialanalyse ein?

Weiteres Feedback

- Was würden Sie sich noch mehr wünschen? Um die Aspekte von vorhin zu adressieren?
- Hintergrundinformation

B.2 Results of *Agile Potential Analysis* Validation

B.2.1 Results of Capgemini Case Study

Table 29: Results of Capgemini case study

	Questions		Capg.1	Capg.2	Capg.3	Capg.4	Capg.5	Capg.6	Capg.7	Capg.8	Capg.9
		ID	1	3	2	4	7	8	9	10	12
		Role	ACA Executor	Process Owner	Process User	Process User	Process User	Process User	Process User	Process User	Process User
Goal Achievement	**… select appropriate Imp. Actions**	1.1	4	3	5	4	4	4	4	3	4
	… achieving our Imp. Goals	1.2	4	4	4	3	3	3	4	NA	NA
	… product quality	1.2.1	3	3	NA	3	3	3	4	NA	3
	… flexibility	1.2.2	3	3	NA	3	3	3	3	NA	4
	… customer collaboration	1.2.3	3	3	NA	3	3	3	3	NA	3
	… knowledge transfer	1.2.4	4	4	NA	4	4	3	4	NA	4
	… transparency	1.2.5	4	4	4	4	4	3	4	4	4
	… achieving SPI goals	1.3	4	3	4	4	3	3	4	NA	NA
	… new improvement goals	1.4	4	3	4	4	4	3	3	NA	NA
	MEDIAN		*4*	*3*	*4*	*4*	*3,5*	*3*	*4*	*3*	*4*
Understandability	**Purpose**	4	4	2	5	4	5	4	4	4	4
	Process	5	3	2	3	4	3	1	3	NA	2
	… clear in meaning	6.1	4	2	5	NA	4	4	4	NA	2
	… easy to comprehend	6.2	4	NA	3	NA	3	2	4	NA	3
	… easy to read	6.3	4	2	3	NA	4	2	4	NA	3
	… in general understandable	6.4	4	3	4	NA	5	3	4	NA	4
	MEDIAN		*4*	*2*	*3,5*	*NA*	*4*	*2,5*	*4*	*NA*	*3*
Reliability	**… trustworthy**	7.1	4	NA	3	3	4	4	4	4	NA
	… accurate	7.2	5	NA	4	3	3	4	4	2	2
	… credible	7.3	4	NA	4	3	4	4	4	NA	NA
	… in general reliable	7.4	4	NA	5	3	3	4	4	NA	2
	MEDIAN		*4*	*NA*	*4*	*3*	*3,5*	*4*	*4*	*3*	*2*
Transparency	**… useful to our work**	8	4	3	4	4	4	3	4	NA	4
	… relevant to our work	9	NA	2	4	4	4	4	4	NA	4
	… appropriate for our work	10	NA	2	2	3	3	2	3	NA	2
	… applicable to our work	11	4	2	3	4	2	2	4	NA	3
	MEDIAN		*4*	*2*	*3,5*	*4*	*3,5*	*2,5*	*4*	*NA*	*3,5*

Usability	**… improves my performance**	12	NA	2	2	3	4	2	4	3	3
	… increases my productivity	13	NA	3	2	3	3	2	4	3	2
	… enhances my effectiveness	14	NA	3	2	3	4	2	4	3	3
	… useful in my job	15	NA	3	4	3	3	3	4	3	3
	MEDIAN		*NA*	*3*	*2*	*3*	*3,5*	*2*	*4*	*3*	*3*
Ease of Use	**… clear and understandable**	16	4	2	4	4	4	2	4	NA	NA
	… not require a lot of my mental effort	17	4	NA	4	4	3	4	4	4	NA
	… easy to use	18	4	3	NA	4	3	3	4	NA	NA
	… do what I want to do	19	4	3	3	4	3	2	4	NA	NA
	MEDIAN		*4*	*3*	*4*	*4*	*3*	*2,5*	*4*	*4*	*NA*
Job relevancy	**In my job … important**	20	NA	3	2	3	4	3	4	NA	3
	In my job … relevant	21	NA	2	4	3	4	4	4	NA	2
	… job-related tasks	22	NA	NA	2	3	4	2	3	NA	1
	MEDIAN		*NA*	*2,5*	*2*	*3*	*4*	*3*	*4*	*NA*	*2*
Outcome quality	**The quality of the output … is high**	23	3	3	4	3	3	2	3	NA	3
	I have no problem with the quality	24	4	2	4	4	4	2	4	NA	3
	I rate the results to be ex-cellent	25	3	3	3	3	3	2	3	NA	2
	MEDIAN		*3*	*3*	*4*	*3*	*3*	*2*	*3*	*NA*	*3*
Outcome	**No difficulties in telling …**	26	4	4	5	4	4	4	3	NA	2
	I could communicate to others …	27	4	4	5	4	4	4	3	NA	1
	Results are apparent to me	28	4	3	4	4	4	4	4	NA	2
	Difficulties in explaining … being beneficial	29	4	3	2	2	4	3	3	NA	2
	MEDIAN		*4*	*3,5*	*4,5*	*4*	*4*	*4*	*3*	*NA*	*2*

B.2.2 Results of VSF Experts Case Study

Table 30: Results of VSF Experts case study

	Questions		VSF.1	VSF.2
		ID	1	2
		Role	ACA Executor	Process User
Goal Achievement	**… select appropriate Imp. Actions**	1.1	4	4
	… achieving our Imp. Goals	1.2	4	3
	… MVP	1.2.1	4	4
	… knowledge transfer	1.2.2	3	3
	… roll-outs	1.2.3	3	3
	… productivity	1.2.4	NA	3
	… planning	1.2.5	NA	4
	… transparency	1.2.6	4	3
	… communication	1.2.7	3	4
	… commitment	1.2.5	3	3
	… achieving SPI goals	1.3	4	3
	… new improvement goals	1.4	4	4
	MEDIAN		4	3,5
Understandability	**Purpose**	4	5	4
	Process	5	4	4
	… clear in meaning	6.1	4	3
	… easy to comprehend	6.2	4	3
	… easy to read	6.3	4	4
	… in general understandable	6.4	4	2
	MEDIAN		4	3,5
Reliability	**… trustworthy**	7.1	3	4
	… accurate	7.2	4	4
	… credible	7.3	3	3
	… in general reliable	7.4	4	3
	MEDIAN		3,5	3,5
Transparency	**… useful to our work**	8	4	4
	… relevant to our work	9	5	3
	… appropriate for our work	10	4	4
	… applicable to our work	11	4	5
	MEDIAN		4	4
Usability	**… improves my performance**	12	NA	4
	… increases my productivity	13	NA	3

	… enhances my effectiveness	14	NA	3
	… useful in my job	15	5	4
	MEDIAN		5	3,5
Ease of Use	**… clear and understandable**	16	4	4
	… not require a lot of my mental effort	17	2	2
	… easy to use	18	3	2
	… do what I want to do	19	2	3
	MEDIAN		2,5	2,5
Job relevancy	**In my job … important**	20	4	3
	In my job … relevant	21	4	4
	… job-related tasks	22	5	4
	MEDIAN		4	4
Outcome quality	**The quality of the output … is high**	23	4	NA
	I have no problem with the quality	24	3	3
	I rate the results to be excellent	25	3	3
	MEDIAN		3	3
Outcome	**No difficulties in telling …**	26	4	5
	I could communicate to others …	27	5	5
	Results are apparent to me	28	NA	4
	Difficulties in explaining … being beneficial	29	5	4
	MEDIAN		5	4,5

B.2.3 Results of Yatta Case Study

Table 31: Results of Yatta case study

	Questions		Y.1	Y.2
		ID	FS	AS
		Role	Process User	Process Owner
Goal Achievement	**… select appropriate Imp. Actions**	1.1	4	4
	… achieving our Imp. Goals	1.2	4	4
	… time to market	**1.2.1**	NA	4
	… difficultied to recieve feedback	**1.2.2**	5	3
	… requirements are vague	**1.2.3**	NA	5
	… knowedge transfer	**1.2.4**	5	5
	… achieving SPI goals	1.3	5	4
	… new improvement goals	1.4	4	4
	MEDIAN		**4**	**4**
Understandability	**Purpose**	4	5	5
	Process	5	5	4
	… clear in meaning	6.1	4	3
	… easy to comprehend	6.2	4	2
	… easy to read	6.3	5	2
	… in general understandable	6.4	5	3
	MEDIAN		**5**	**3**
Reliability	**… trustworthy**	7.1	5	4
	… accurate	7.2	5	4
	… credible	7.3	5	4
	… in general reliable	7.4	4	4
	MEDIAN		**5**	**4**
Transparency	**… useful to our work**	8	5	3
	… relevant to our work	9	5	4
	… appropriate for our work	10	4	3
	… applicable to our work	11	4	3
	MEDIAN		**4,5**	**3**

Usability	**… improves my performance**	12	5	3
	… increases my productivity	13	5	3
	… enhances my effectiveness	14	4	3
	… useful in my job	15	4	4
	MEDIAN		**4,5**	**3**
Ease of Use	**… clear and understandable**	16	5	2
	… not require a lot of my mental effort	17	4	2
	… easy to use	18	4	2
	… do what I want to do	19	5	2
	MEDIAN		**4,5**	**2**
Job relevancy	**In my job … important**	20	4	3
	In my job … relevant	21	5	4
	… job-related tasks	22	4	3
	MEDIAN		**4**	**3**
Outcome quality	**The quality of the output … is high**	23	5	2
	I have no problem with the quality	24	5	2
	I rate the results to be excellent	25	5	2
	MEDIAN		**5**	**2**
Outcome	**No difficulties in telling …**	26	5	3
	I could communicate to others …	27	4	4
	Results are apparent to me	28	5	4
	Difficulties in explaining … being beneficial	29	4	4
	MEDIAN		**4,5**	**4**

B.2.4 Results of Kemweb Case Study

Table 32: Results of Kemweb case study

	Questions		KW.1	KW.2	KW.3	KW.4	KW.5	KW.6	KW.7
		ID	1435	1436	1437	1438	CW	MAK	1441
		Role	Process User	Process User	APA Executor, Process Owner	Process User	Process User	Process User	Process User
Goal Achievement	**… select appropriate Imp. Actions**	1.1	5	4	5	5	5	4	4
	… achieving our Imp. Goals	1.2	4	3	4	4	4	NA	3
	… project transparency	**1.2.1**	NA	4	5	4	4	NA	3
	… translation of wished to requirements	**1.2.2**	4	3	5	3	5	NA	4
	… communication	**1.2.3**	NA	4	5	5	3	NA	4
	… knowedge transfer	**1.2.4**	3	NA	4	4	3	NA	4
	… understandability by the customer	**1.2.5**	3	NA	3	3	4	NA	3
	… documentation	**1.2.6**	4	3	4	2	3	NA	3
	… achieving SPI goals	1.3	4	4	4	4	5	NA	4
	… new improvement goals	1.4	NA	3	5	4	5	NA	4
	MEDIAN		4	3,5	4,5	4	5	4	4
Understandability	**Purpose**	4	4	2	5	4	4	5	3
	Process	5	3	2	3	5	4	4	2
	… clear in meaning	6.1	4	4	5	4	4	4	3
	… easy to comprehend	6.2	4	3	3	5	4	4	4
	… easy to read	6.3	3	3	3	3	4	4	4
	… in general understandable	6.4	4	4	5	5	4	4	3
	MEDIAN		4	3	4	4,5	4	4	3
Reliability	**… trustworthy**	7.1	4	4	5	4	5	5	4
	… accurate	7.2	4	4	5	4	4	5	3
	… credible	7.3	4	3	4	4	4	5	4
	… in general reliable	7.4	3	4	5	4	4	4	5
	MEDIAN		4	4	5	4	4	5	4
Transparency	**… useful to our work**	8	4	4	5	4	5	4	4
	… relevant to our work	9	4	3	4	4	5	4	5
	… appropriate for our work	10	3	4	5	4	4	4	4
	… applicable to our work	11	3	4	4	4	4	3	4
	MEDIAN		3,5	4	4,5	4	4,5	4	4

Usability	**… improves my performance**	12	2	4	5	5	4	NA	4
	… increases my productivity	13	4	5	4	4	4	NA	4
	… enhances my effectiveness	14	3	4	4	3	4	NA	4
	… useful in my job	15	3	4	5	5	4	4	4
	MEDIAN		3	4	4,5	4,5	4	4	4
Ease of Use	**… clear and understandable**	16	3	3	4	3	3	4	3
	… not require a lot of my mental effort	17	3	2	2	1	2	2	3
	… easy to use	18	3	2	2	3	4	4	3
	… do what I want to do	19	3	3	2	3	3	3	3
	MEDIAN		3	2,5	2	3	3	3,5	3
Job relevancy	**In my job … important**	20	3	4	5	5	4	4	4
	In my job … relevant	21	3	3	4	4	4	4	4
	… job-related tasks	22	3	3	4	4	4	3	4
	MEDIAN		3	3	4	4	4	4	4
Outcome quality	**The quality of the output … is high**	23	3	4	4	3	4	4	3
	I have no problem with the quality	24	3	5	4	5	4	4	3
	I rate the results to be excellent	25	3	3	3	3	4	4	3
	MEDIAN		3	4	4	3	4	4	3
Outcome	**No difficulties in telling …**	26	3	4	5	4	4	5	3
	I could communicate to others …	27	3	4	5	4	4	4	3
	Results are apparent to me	28	3	4	5	3	4	5	4
	Difficulties in explaining … being beneficial	29	3	3	5	4	3	5	4
	MEDIAN		3	4	5	4	4	5	3,5

B.2.5 Results of TQsoft Case Study

Table 33: Results of TQsoft case study

	Questions		TQ.1	TQ.2	TQ.3	TQ.4	TQ.5
		ID	YR	UL	TL	RV	JW
		Role	Process Owner	Process User	Process User	Process User	Process User
Goal Achievement	**… select appropriate Imp. Actions**	1.1	4	4	4	4	4
	… achieving our Imp. Goals	1.2	4	4	4	4	4
	… documentation	**1.2.1**	5	5	3	4	4
	… higher QA-effort	**1.2.2**	3	NA	3	3	3
	… improve project approval	**1.2.3**	3	5	4	3	3
	… SW architecture	**1.2.4**	3	4	NA	3	NA
	… achieving SPI goals	1.3	4	5	4	4	5
	… new improvement goals	1.4	4	4	4	4	3
	MEDIAN		**4**	**4**	**4**	**4**	**4**
Understandability	**Purpose**	4	5	4	4	5	4
	Process	5	4	3	4	5	5
	… clear in meaning	6.1	5	5	4	3	4
	… easy to comprehend	6.2	5	3	4	3	3
	… easy to read	6.3	5	3	5	2	2
	… in general understandable	6.4	5	4	5	3	4
	MEDIAN		**5**	**3,5**	**4**	**3**	**4**
Reliability	**… trustworthy**	7.1	5	3	5	5	4
	… accurate	7.2	5	4	4	4	3
	… credible	7.3	5	3	4	4	4
	… in general reliable	7.4	5	4	4	4	4
	MEDIAN		**5**	**3,5**	**4**	**4**	**4**
Transparency	**… useful to our work**	8	4	4	4	4	5
	… relevant to our work	9	5	5	4	3	5
	… appropriate for our work	10	4	4	4	4	4
	… applicable to our work	11	4	4	4	4	4
	MEDIAN		**4**	**4**	**4**	**4**	**4,5**

Usability	**… improves my performance**	12	4	5	3	4	3
	… increases my productivity	13	3	4	NA	3	3
	… enhances my effectiveness	14	4	4	3	3	4
	… useful in my job	15	4	4	4	4	5
	MEDIAN		**4**	**4**	**3**	**3,5**	**3,5**
Ease of Use	**… clear and understandable**	16	4	NA	3	3	4
	… not require a lot of my mental effort	17	2	2	2	3	3
	… easy to use	18	4	2	3	2	3
	… do what I want to do	19	3	3	3	3	NA
	MEDIAN		**3,5**	**2**	**3**	**3**	**3**
Job relevancy	**In my job … important**	20	5	4	4	2	4
	In my job … relevant	21	4	4	4	3	5
	… job-related tasks	22	3	3	4	2	4
	MEDIAN		**4**	**4**	**4**	**2**	**4**
Outcome quality	**The quality of the output … is high**	23	4	4	4	4	4
	I have no problem with the quality	24	3	4	4	3	4
	I rate the results to be excellent	25	3	4	3	3	3
	MEDIAN		**3**	**4**	**4**	**3**	**4**
Outcome	**No difficulties in telling …**	26	5	2	4	4	5
	I could communicate to others …	27	5	3	4	3	4
	Results are apparent to me	28	5	3	4	4	3
	Difficulties in explaining … being beneficial	29	1	2	2	3	4
	MEDIAN		**5**	**2,5**	**4**	**3,5**	**4**

B.2.6 Results of Bosch Case Study

Table 34: Results of Bosch case study

	Questions		CAP.1
		ID	1
		Role	Process User
Goal Achievement	**… select appropriate Imp. Actions**	1.1	NA
	… achieving our Imp. Goals	1.2	NA
	… product quality	1.2.1	NA
	… flexibility	1.2.2	NA
	… customer collaboration	1.2.3	NA
	… knowledge transfer	1.2.4	NA
	… transparency	1.2.5	NA
	… achieving SPI goals	1.3	NA
	… new improvement goals	1.4	NA
	MEDIAN		**NA**
Understandability	**Purpose**	4	4
	Process	5	4
	… clear in meaning	6.1	4
	… easy to comprehend	6.2	3
	… easy to read	6.3	4
	… in general understandable	6.4	4
	MEDIAN		**4**
Reliability	**… trustworthy**	7.1	4
	… accurate	7.2	3
	… credible	7.3	4
	… in general reliable	7.4	4
	MEDIAN		**4**
Transparency	**… useful to our work**	8	4
	… relevant to our work	9	4
	… appropriate for our work	10	4
	… applicable to our work	11	4
	MEDIAN		**4**

Usability	**… improves my performance**	12	3
	… increases my productivity	13	3
	… enhances my effectiveness	14	4
	… useful in my job	15	4
	MEDIAN		**3,5**
Ease of Use	**… clear and understandable**	16	3
	… not require a lot of my mental effort	17	3
	… easy to use	18	3
	… do what I want to do	19	3
	MEDIAN		**3**
Job relevancy	**In my job … important**	20	2
	In my job … relevant	21	3
	… job-related tasks	22	2
	MEDIAN		**2**
Outcome quality	**The quality of the output … is high**	23	4
	I have no problem with the quality	24	3
	I rate the results to be excellent	25	3
	MEDIAN		**3**
Outcome	**No difficulties in telling …**	26	4
	I could communicate to others …	27	3
	Results are apparent to me	28	3
	Difficulties in explaining … being beneficial	29	2
	MEDIAN		**3**

B.3 Results of Simulation Validation

This section of the appendix contains the individual results of the simulation validation. For the sake of simplicity and transparency, we provide the original German results.

B.3.1 Results of Capgemini Interview

Interview zum Simulations-AddOn der Agilen Potenzialanalyse

Einschätzungen im Vergleich zur Agilen Potenzialanalyse

- Wie schätzen Sie dieses AddOn im Vergleich zur Potenzialanalyse generell ein?
- Welchem Mehrwert sehen Sie in diesem AddOn im Vergleich zur Potenzialanalyse?
    - Erster Eindruck: „total gut"

Verständnis (=Comprehension)

- Wie schätzen Sie die Verständlichkeit des AddOn im Vergleich zur Potenzialanalyse ein?
    - Es ist direkt zusehen was die Auswirkungen sind
    - Das Zustandekommen der Ergebnisse ist (jetzt) klar und die Visualisierung durch die erklärten Schritte sind verständlich
- Wie schätzen Sie die Transparenz des AddOn im Vergleich zur Potenzialanalyse ein?
    - Viel transparenter und damit ein großer Vorteil
    - Vorher welche Ziele gut sind (mit Algorithmus) → vorher bemuttert
      → Nun kann man die Ziele Vergleichen und Abwägen über konkrete Folgen
    - Man sieht was berechnet (z.B. „addiert") wird und muss dennoch nicht alle mathematischen Details verstehen
- Wie schätzen Sie die Zuverlässigkeit des AddOn im Vergleich zur Potenzialanalyse ein?
    - Sehr interessant bzw. Nachvollziehbarkeit
    - Mehr selbst entscheiden können macht das AddOn zu einem Mehrwert
        - „Nur Ergebnisse vorgesetzt bekommen, dann traue ich per se nicht"

Akzeptanz (=Acceptance)

- Wie schätzen Sie die Relevanz des AddOn im Vergleich zur Potenzialanalyse ein?
    - Wofür relevant? Bestimmte Entscheidungen? ...?
- Wie schätzen Sie das Ergebnis des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Qualität des Ergebnisses des AddOn im Vergleich zur Potenzialanalyse ein?

- „leichtes Plus"
    - Kommt drauf an, wie man das Ergebnis in das Projekt bzw. Unternehmen bringt
      → mit Präsentation und Durchgehen ist das ein „großes Plus" sonst wäre es bei „Null" Akzeptanz
- „An dieser Stelle brauche ich dann keine weiteren mathematischen Details"
- Interpretation natürlich immer noch notwendig

Weiteres Feedback

- Speziell mit dem AddOn ist es eine gute Grundlage um mit anderen Personen zu diskutieren
    - Würde die Ergebnisse bzw. die Methode gerne innerhalb von Capgemini verbreiten
- Nimmt auf jeden Fall den richtigen Weg → Mit der Weiterentwicklung ist es ein super Update

B.3.2 Results of Kemweb Interview

Interview zum Simulations-AddOn der Agilen Potenzialanalyse

Einschätzungen im Vergleich zur Agilen Potenzialanalyse

- Wie schätzen Sie dieses AddOn im Vergleich zur Potenzialanalyse generell ein?
- Welchem Mehrwert sehen Sie in diesem AddOn im Vergleich zur Potenzialanalyse?
  - Dieses AddOn ist schon mal sicher ein Mehrwert
  - „Wie kommen wir auf diese Daten?"
    (siehe letzten Punkt der Antworten auf diese Frage)
  - Zusammenfassung:
    Ziele die definiert sind, die ich verbessern will und Bausteine wirken darauf hin
    → „Nächster logischer Schritt ist Gewichtung bzw. Bewertung durch die Daten"
  - „Find ich sehr gut"
  - „Ich habe nicht genau verstanden wie es sich berechnet. Muss ich aber vermutlich auch nicht."

Verständnis (=Comprehension)

- Wie schätzen Sie die Verständlichkeit des AddOn im Vergleich zur Potenzialanalyse ein?
  - Mein Problem dazu was auszusagen, da Verständlichkeit der Grafiken sehr komplex
  - Erklärung ist notwendig → eher schlechte Verständlichkeit, da „mehr" als vorher
  - Ja, Mehrwert (Details) ist auch Zeit um es zu verstehen wert
  - Wären die Grafiken/Diagramme in dem Excel verständlicher?
    - Kann ich nach der Erklärung nun nicht mehr sagen, da auch dies jetzt verständlich ist
- Wie schätzen Sie die Transparenz des AddOn im Vergleich zur Potenzialanalyse ein?
  - „Detailstufe" = Transparenz erhöht
- Wie schätzen Sie die Zuverlässigkeit des AddOn im Vergleich zur Potenzialanalyse ein?
  - „Ich find es nicht zuverlässiger aber detaillierter und genauer?"
  - „als Empfehlung"

Akzeptanz (=Acceptance)

- Wie schätzen Sie die Relevanz des AddOn im Vergleich zur Potenzialanalyse ein?
  - Wofür relevant? Bestimmte Entscheidungen? ...?
- Wie schätzen Sie das Ergebnis des AddOn im Vergleich zur Potenzialanalyse ein?
  - Ergebnis schön sehen, da mehr Details und jetzt wird der „Zonk" konkreter definiert
    (bezieht sich konkret auf das Beispiel, BurnChart als „Zonk")
  - Wissenschaftlich bzw. mathematisch fundiert
  - Visuelle bzw. Verständnis Sache → Transfer wird durch das AddOn abgenommen
- Wie schätzen Sie die Qualität des Ergebnisses des AddOn im Vergleich zur Potenzialanalyse ein?
  - *Siehe vorherige Frage*
- Wie schätzen Sie die Benutzbarkeit des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Bedienkomfort des AddOn im Vergleich zur Potenzialanalyse ein?

Weiteres Feedback

- Was würden Sie sich noch mehr wünschen? Um die Aspekte von vorhin zu adressieren?
- Hintergrundinformation
  - „eher so ein Tabellen-Typ"
  - „kannst die Repräsentation von Folie 2 schon"

B.3.3 Results of TQsoft Interview 1

Interview zum Simulations-AddOn der Agilen Potenzialanalyse

Einschätzungen im Vergleich zur Agilen Potenzialanalyse

- Wie schätzen Sie dieses AddOn im Vergleich zur Potenzialanalyse generell ein?
- Welchem Mehrwert sehen Sie in diesem AddOn im Vergleich zur Potenzialanalyse?
    - Neues Feature
        - Risikobetrachtung
        - Aufzeigen weiterer Potenziale (bzw. Hinterfragen mancher Aspekte):
            - „Wo auf dem schwarzen Balken befinde ich mich schon?" und „Wie gut funktioniert es schon?" → Damit „Wo könnte ich noch hin?"

Verständnis (=Comprehension)

- Wie schätzen Sie die Verständlichkeit des AddOn im Vergleich zur Potenzialanalyse ein?
    - Verwirrend ist, dass es eine 0,0 bei den Diagrammen in Folie 9 gibt
    - Bewusst weg vom Balken
    - Auch das Excel bedarf schon Erklärung und ist sehr komplex. Ist hier einfacher, was natürlich auch an der Reduktion der Informationen (3 x 3) liegen könnte
- Wie schätzen Sie die Transparenz des AddOn im Vergleich zur Potenzialanalyse ein?
    - „was noch nach oben gehen könnte"
- Wie schätzen Sie die Zuverlässigkeit des AddOn im Vergleich zur Potenzialanalyse ein?
    -

Akzeptanz (=Acceptance)

- Wie schätzen Sie die Relevanz des AddOn im Vergleich zur Potenzialanalyse ein?
    - Wofür relevant? Bestimmte Entscheidungen? ...?
- Wie schätzen Sie das Ergebnis des AddOn im Vergleich zur Potenzialanalyse ein?
    - „Ja, die Akzeptanz ist bei mir schon höher damit", aber
    - Es kommt drauf an, wer im Unternehmen wie denkt
    - Kann damit vermutlich keinen reinen Entwickler begeistern → bisschen zu abstrakt
        - Was könnte ihn begeistern?
          Abbildung auf konkrete/direkte Ziele bzw. konkreten Alltag
            - Folie 8: Figur an konkreten Probleme könnte noch sinnvoller sein für die Personen die diese genannt haben
    - Mehr Akzeptanz, wenn ich sehe wo er sich überall gut auswirkt
- Wie schätzen Sie die Qualität des Ergebnisses des AddOn im Vergleich zur Potenzialanalyse ein?
    - *Siehe vorherige Frage*
- Wie schätzen Sie die Benutzbarkeit des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Bedienkomfort des AddOn im Vergleich zur Potenzialanalyse ein?

Weiteres Feedback

- Was würden Sie sich noch mehr wünschen? Um die Aspekte von vorhin zu adressieren?
    - Ggf. nur plus und minus als Visualisierung und nicht die vier Kategorien
- Gut vorstellbar: „Wo bekomme ich die Infos her wo die Probleme liegen?"
    - WS / Folie mit den Themen: Abholen der Personen
- „Sind die schon verwendeten Praktiken nicht mehr hier im Bild?"
  → ggf. wäre die Idee des taggen der verwendeten Praktiken am besten
  (ähnlich zu Filter-Mechanismus in Excel)

B.3.4 Results of TQsoft Interview 2

Interview zum Simulations-AddOn der Agilen Potenzialanalyse

Einschätzungen im Vergleich zur Agilen Potenzialanalyse

- Wie schätzen Sie dieses AddOn im Vergleich zur Potenzialanalyse generell ein?
- Welchem Mehrwert sehen Sie in diesem AddOn im Vergleich zur Potenzialanalyse?
    - Gefühlt erstmal erschlagen werden
    - Überlegung: Wie bekommt man das noch weiter eingegrenzt?
    - Geht nur mit Erklärung
    - Interpretation muss gelernt sein

Verständnis (=Comprehension)

- Wie schätzen Sie die Verständlichkeit des AddOn im Vergleich zur Potenzialanalyse ein?
    - Schön visualisiert
    - Wahnsinnig viel Informationen draufgepresst
- Wie schätzen Sie die Transparenz des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Zuverlässigkeit des AddOn im Vergleich zur Potenzialanalyse ein?
- Interaktiv wäre es sinnvoller → Stück für Stück aufbauen mit immer mehr Icons oder detaillierte Erläuterung
    - Wenn man es Stück für Stück aufbauen, bekommt man jemanden gut ran geführt → erhöht Verständnis
- Ziel muss es sein, damit „Vertrauensgefühl" abzugeben
- Besser als klassische Excel-Listen → Excel Listen „brauchen ja gar nicht so viel Aufwand, da muss man nur Zahlen reinhacken"
- Psychologisch mehr Eindrücke → „bin zu viel mehr möglich" → deutlich höheres Vertrauen → Thema weiter ausbauen
- „Visualisierung ist die Kür auf die (Excel-)Liste"

Akzeptanz (=Acceptance)

- Wie schätzen Sie die Relevanz des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie das Ergebnis des AddOn im Vergleich zur Potenzialanalyse ein?
    - Art und Weise der Visualisierung hinterlässt mehr eintrug und damit wirkt das Ergebnisse Nachhaltiger
- Wie schätzen Sie die Qualität des Ergebnisses des AddOn im Vergleich zur Potenzialanalyse ein?
    - Art und Weise der Visualisierung hinterlässt mehr eintrug und damit wirkt das Ergebnisse Nachhaltiger
- Wie schätzen Sie die Benutzbarkeit des AddOn im Vergleich zur Potenzialanalyse ein?
- Wie schätzen Sie die Bedienkomfort des AddOn im Vergleich zur Potenzialanalyse ein?

Weiteres Feedback

- Excel mit interaktiv würde helfen

Curriculum Vitae

Name	Philipp Diebold	
Anschrift	Dammweg 15 69123 Heidelberg	
Geburtsdatum	01.04.1988	
Geburtsort	München	
Staatsangehörigkeit	Deutsch	
Schulbildung	1994 - 1998	Grundschule
	1998 - 2007	Gymnasium Abschluss: Abitur
Studium	2007 - 2012	Informatik Technische Universität Kaiserslautern, Kaiserslautern, Deutschland Abschluss: Master of Science, Informatik
Berufstätigkeit	2012 - 2015	Wissenschaftlicher Mitarbeiter Fraunhofer IESE, Kaiserslautern
	2015 - 2018	Projektleiter Fraunhofer IESE, Kaiserslautern
	2018 - heute	Geschäftsführer Bagilstein GmbH

Heidelberg, den 9.12.2019

PhD Theses in Experimental Software Engineering

Volume 1 **Oliver Laitenberger** (2000), *Cost-Effective Detection of Software Defects Through Perspective-based Inspections*

Volume 2 **Christian Bunse** (2000), *Pattern-Based Refinement and Translation of Object-Oriented Models to Code*

Volume 3 **Andreas Birk** (2000), *A Knowledge Management Infrastructure for Systematic Improvement in Software Engineering*

Volume 4 **Carsten Tautz** (2000), *Customizing Software Engineering Experience Management Systems to Organizational Needs*

Volume 5 **Erik Kamsties** (2001), *Surfacing Ambiguity in Natural Language Requirements*

Volume 6 **Christiane Differding** (2001), *Adaptive Measurement Plans for Software Development*

Volume 7 **Isabella Wieczorek** (2001), *Improved Software Cost Estimation A Robust and Interpretable Modeling Method and a Comprehensive Empirical Investigation*

Volume 8 **Dietmar Pfahl** (2001), *An Integrated Approach to Simulation-Based Learning in Support of Strategic and Project Management in Software Organisations*

Volume 9 **Antje von Knethen** (2001), *Change-Oriented Requirements Traceability Support for Evolution of Embedded Systems*

Volume 10 **Jürgen Münch** (2001), *Muster-basierte Erstellung von Software-Projektplänen*

Volume 11 **Dirk Muthig** (2002), *A Light-weight Approach Facilitating an Evolutionary Transition Towards Software Product Lines*

Volume 12 **Klaus Schmid** (2003), *Planning Software Reuse – A Disciplined Scoping Approach for Software Product Lines*

Volume 13 **Jörg Zettel** (2003), *Anpassbare Methodenassistenz in CASE-Werkzeugen*

Volume 14 **Ulrike Becker-Kornstaedt** (2004), *Prospect: a Method for Systematic Elicitation of Software Processes*

Volume 15 **Joachim Bayer** (2004), *View-Based Software Documentation*

Volume 16 **Markus Nick** (2005), *Experience Maintenance through Closed-Loop Feedback*

Volume 17 **Jean-François Girard** (2005), *ADORE-AR: Software Architecture Reconstruction with Partitioning and Clustering*

Volume 18 **Ramin Tavakoli Kolagari** (2006), *Requirements Engineering für Software-Produktlinien eingebetteter, technischer Systeme*

Volume 19 **Dirk Hamann** (2006), *Towards an Integrated Approach for Software Process Improvement: Combining Software Process Assessment and Software Process Modeling*

Volume 20 **Bernd Freimut** (2006), *MAGIC: A Hybrid Modeling Approach for Optimizing Inspection Cost-Effectiveness*

Volume 21 **Mark Müller** (2006), *Analyzing Software Quality Assurance Strategies through Simulation. Development and Empirical Validation of a Simulation Model in an Industrial Software Product Line Organization*

Volume 22 **Holger Diekmann** (2008), *Software Resource Consumption Engineering for Mass Produced Embedded System Families*

Volume 23 **Adam Trendowicz** (2008), *Software Effort Estimation with Well-Founded Causal Models*

Volume 24 **Jens Heidrich** (2008), *Goal-oriented Quantitative Software Project Control*

Volume 25 **Alexis Ocampo** (2008), *The REMIS Approach to Rationale-based Support for Process Model Evolution*

Volume 26 **Marcus Trapp** (2008), *Generating User Interfaces for Ambient Intelligence Systems; Introducing Client Types as Adaptation Factor*

Volume 27 **Christian Denger** (2009), *SafeSpection – A Framework for Systematization and Customization of Software Hazard Identification by Applying Inspection Concepts*

Volume 28 **Andreas Jedlitschka** (2009), *An Empirical Model of Software Managers' Information Needs for Software Engineering Technology Selection*
A Framework to Support Experimentally-based Software Engineering Technology Selection

Volume 29 **Eric Ras** (2009), *Learning Spaces: Automatic Context-Aware Enrichment of Software Engineering Experience*

Volume 30 **Isabel John** (2009), *Pattern-based Documentation Analysis for Software Product Lines*

Volume 31 **Martín Soto** (2009), *The DeltaProcess Approach to Systematic Software Process Change Management*

Volume 32 **Ove Armbrust** (2010), *The SCOPE Approach for Scoping Software Processes*

Volume 33 **Thorsten Keuler** (2010), *An Aspect-Oriented Approach for Improving Architecture Design Efficiency*

Volume 34 **Jörg Dörr** (2010), *Elicitation of a Complete Set of Non-Functional Requirements*

Volume 35 **Jens Knodel** (2010), *Sustainable Structures in Software Implementations by Live Compliance Checking*

Volume 36 **Thomas Patzke** (2011), *Sustainable Evolution of Product Line Infrastructure Code*

Volume 37 **Ansgar Lamersdorf** (2011), *Model-based Decision Support of Task Allocation in Global Software Development*

Volume 38 **Ralf Carbon** (2011), *Architecture-Centric Software Producibility Analysis*

Volume 39 **Florian Schmidt** (2012), *Funktionale Absicherung kamerabasierter Aktiver Fahrerassistenzsysteme durch Hardware-in the-Loop-Tests*

Volume 40 **Frank Elberzhager** (2012), *A Systematic Integration of Inspection and Testing Processes for Focusing Testing Activities*

Volume 41 **Matthias Naab** (2012), *Enhancing Architecture Design Methods for Improved Flexibility in Long-Living Information Systems*

Volume 42 **Marcus Ciolkowski** (2012), *An Approach for Quantitative Aggregation of Evidence from Controlled Experiments in Software Engineering*

Volume 43 **Igor Menzel** (2012), *Optimizing the Completeness of Textual Requirements Documents in Practice*

Volume 44 **Sebastian Adam** (2012), *Incorporating Software Product Line Knowledge into Requirements Processes*

Volume 45 **Kai Höfig** (2012), *Failure-Dependent Timing Analysis – A New Methodology for Probabilistic Worst-Case Execution Time Analysis*

Volume 46 **Kai Breiner** (2013), *AssistU – A framework for user interaction forensics*

Volume 47 **Rasmus Adler** (2013), *A model-based approach for exploring the space of adaptation behaviors of safety-related embedded systems*

Volume 48 **Daniel Schneider** (2014), *Conditional Safety Certification for Open Adaptive Systems*

Volume 49 **Michail Anastasopoulos** (2013), *Evolution Control for Software Product Lines: An Automation Layer over Configuration Management*

Volume 50 **Bastian Zimmer** (2014), *Efficiently Deploying Safety-Critical Applications onto Open Integrated Architectures*

Volume 51 **Slawomir Duszynski** (2015), *Analyzing Similarity of Cloned Software Variants using Hierarchical Set Models*

Volume 52 **Zhensheng Guo** (2015), *Safe Requirements Engineering: A Scenario-based Approach for Identifying Complete Safety-oriented Requirements*

Volume 53 **Bo Zhang** (2015), *VITAL – Reengineering Variability Specifications and Realizations in Software Product Lines*

Volume 54 **Norman Riegel** (2016), *Prioritization in Incremental Requirements Engineering*

Volume 55 **Pablo Oliveira Antonino de Assis** (2016), *Improving the Consistency and Completeness of Safety Requirements Specifications*

Volume 56 **Thomas Bauer** (2016), *Enabling Functional Integration Testing by Using Heterogeneous Models*

Volume 57 **Michael Kläss** (2016), *HyDEEP: Transparent Combination of Measurement and Expert Data for Defect Prediction*

Volume 58 **Liliana Katherine Guzmán Rehbein** (2017), *Empirically-based Method for Performing Qualitative Synthesis in Software Engineering*

Volume 59 **Michael Roth** (2017), *Qualitative Reliability Analysis of Software-Controlled Systems using State/Event Fault Trees*

Volume 60 **Hadil Abukwaik** (2017), *Proactive Support for Conceptual Interoperability Analysis of Software Units*

Volume 61 **Konstantin Holl** (2018), *Quality Assurance for Mobile Business Applications*

Volume 62 **Dominik Rost** (2019), *Task-Specific Architecture Documentation for Developers*

Volume 63 **Christian Jung** (2019), *Context-aware Security*

Volume 64 **Andreas Maier** (2019), *Identification and Specification of Hedonic Quality in User Requirements*

Volume 65 **Philipp Diebold** (2019), *Agile Practice Experience Repository for Process Improvement*

Software Engineering has become one of the major foci of Computer Science research in Kaiserslautern, Germany. Both the University of Kaiserslautern's Computer Science Department and the Fraunhofer Institute for Experimental Software Engineering (IESE) conduct research that subscribes to the development of complex software applications based on engineering principles. This requires system and process models for managing complexity, methods and techniques for ensuring product and process quality, and scalable formal methods for modeling and simulating system behavior. To understand the potential and limitations of these technologies, experiments need to be conducted for quantitative and qualitative evaluation and improvement. This line of software engineering research, which is based on the experimental scientific paradigm, is referred to as 'Experimental Software Engineering'.
In this series, we publish PhD theses from the Fraunhofer Institute for Experimental Software Engineering (IESE) and from the Software Engineering Research Groups of the Computer Science Department at the University of Kaiserslautern. PhD theses that originate elsewhere can be included, if accepted by the Editorial Board.

ISBN 978-3-8396-1555-3
9 783839 615553

Fraunhofer IESE

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

PhD Theses in Experimental Software Engineering