Thesis for The Degree of Doctor of Philosophy

Behavioral Software Engineering – Organizational Change Refocused

Per Lenberg

Division of Software Engineering
Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2018

Behavioral Software Engineering –
Organizational Change Refocused

Per Lenberg

ISBN 978-91-7597-810-9
Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 4491.
ISSN 0346-718X

Technical Report No 163D
Department of Computer Science and Engineering
Division of Software Engineering
Chalmers University of Technology
Gothenburg, Sweden

This thesis has been prepared using LaTeX.
Printed by Chalmers Reproservice,
Gothenburg, Sweden 2018.

"Organizations are communities of human beings,
not collections of human resources."
- Henry Mintzberg

Abstract

Background The development of software is not only a technical endeavor; it is significantly affected by the behaviors of the people involved. Since social scientists have been studying humans for over a century, it is likely that insights they have developed could be used to increase software development effectiveness. There are, nevertheless, indications that software engineering researchers seldom use theories developed and proven within the social sciences. Overall, software engineering research that emphasizes human aspects is still limited compared to studies with technology or process focus.

Objective Given the importance of human aspects in software development, we argue that knowledge from the social sciences should be used more extensively to improve software engineering research. Therefore, the primary objective of our research was to advance software engineering by more profoundly considering humans and their behavior. For in-depth insights into such interdisciplinary research efforts, we chose to explore a specific phenomenon: organizational change. Our secondary objective was thus to create understanding to help improve software companies' organizational change efforts.

Method To address our research objectives, we used a variety of research designs and data collection methods, including literature reviews, surveys, interviews, focus groups, and quantitative analysis of project data. This diversity allowed us to examine phenomena from different perspectives.

Results We provide directions for future research on behavioral aspects of software engineering by outlining the behavioral software engineering (BSE) research area, reviewing contemporary research, and identifying industrial needs. Moreover, our findings suggest that software engineers form their attitudes toward change collectively and according to their teams' social norms, which are governed by their distinct professional identity. Our results also indicate that misalignment of organizational values between groups adversely affects change efforts and overall performance.

Conclusions Our research concludes that in order to effectively manage organizational change efforts, software companies must strengthen their organizational identity and reduce misalignment of organizational values. By providing such concrete advice on how to improve organizational change, our research confirms the usefulness of and need for additional BSE research to create novel and in-depth insights into software engineering phenomena.

Keywords

Software engineering, Organizational change, Social science

Acknowledgment

I am grateful to my parents Bo and Elvi, who always have encouraged me to explore my potential and given me much needed moral support. I am also profoundly thankful to my wife Jessica and my children Ellen and Alexander for being a part of my life and supporting me along this journey. Additionally, I appreciate all the help from my aunt Birgitta and my sister Lena with family: Joakim, Oscar, and Alice.

Moreover, I would like to thank my supervisor Professor Robert Feldt and my co-supervisor Associate Professor Lars Göran Wallgren Tengberg for their invaluable guidance and for expecting nothing less than excellence from me. I also appreciate the direct and insightful feedback provided by my examiner Professor Richard Torkar. My sincere thank also goes to my colleagues Dr. Lucas Gren and Dr. Emil Alegroth for their cooperation and friendship.

I am likewise grateful to everyone at Saab ATM; with a mention to Göran Henningsson, Magnus Lindegren, Martin Magnusson, and Fredrik Lundberg.

Finally, last but by no means least, I would like to thank my friends for putting up with my endless monologues during the past five years on matters related to philosophy, psychology, and technology: Magnus Niklasson, Peter Nilsson, Ulf Ivraeus, Anders Magnusson, Madalina Magnusson, Johan Svahn, Ulrika Svahn, Mathias Kronstrand, Josefin Kronstrand, Henrik Söderström, Johan Kaij, Henri Munthe, Daniel Mellqvist, Josefine Mellqvist, Johan Stefors, Christine Cronvall, Johan Gustafsson, and Anders Hagnestål.

Thanks for all your encouragement!

List of Publications

Appended publications

This thesis is based on the following publications:

[A] P. Lenberg, R. Feldt, and L. G. Wallgren, "Behavioral software engineering: A definition and systematic literature review," *Journal of Systems and Software*, vol. 107, pp. 15–37, 2015

[B] P. Lenberg, R. Feldt, and L. G. Wallgren, "Human factors related challenges in software engineering – an industrial perspective," in *Proceedings of the 8th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2015*, pp. 43–49, IEEE, 2015

[C] P. Lenberg, L. G. Wallgren Tengberg, and R. Feldt, "An initial analysis of software engineers' attitudes towards organizational change," *Empirical Software Engineering*, vol. 22, no. 4, pp. 2179–2205, 2017

[D] P. Lenberg, E. Alegroth, R. Feldt, and L. G. Wallgren, "An initial analysis of differences in software engineers' attitudes towards organizational change," in *Proceedings of the 9th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2016*, pp. 1–7, IEEE, 2016

[E] P. Lenberg and R. Feldt, "Psychological safety and norm clarity in software engineering teams," in *Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2018*, pp. 79–86, ACM, 2018

[F] P. Lenberg, R. Feldt, and L. G. Wallgren Tengberg, "The link between team behavior and organizational change." *In submission to a journal*, 2018

[G] P. Lenberg, R. Feldt, and L. G. Wallgren Tengberg, "Misaligned values in software engineering organizations." *Accepted for publication in Journal of Software: Evolution and Process*, 2018

Other publications

The following publications were published during my PhD studies. They are, however, not appended to this thesis.

[a] P. Lenberg, R. Feldt, and L. G. Wallgren, "Towards a behavioral software engineering," in *Proceedings of the 7th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2014*, pp. 48–55, ACM, 2014

[b] P. Lenberg, R. Feldt, L. G. W. Tengberg, I. Tidefors, and D. Graziotin, "Behavioral software engineering - guidelines for qualitative studies," *CoRR*, vol. abs/1712.08341, 2017

[c] L. Gren and P. Lenberg, "The importance of conflict resolution techniques in autonomous agile teams," in *Proceedings of the First International Workshop on Autonomous Agile Teams, A-TEAMS'18*, 2018

Research Contribution

In all appended publications, the first author was the primary contributor to the research idea, research design, data collection, data analysis, and reporting.

Contents

Chapter 1

Introduction

Researchers have repeatedly acknowledged the importance of human aspects in the development of software [11–16]. To improve software engineering effectiveness, researchers must not only consider its technical and process-related concerns but also comprehend in detail the factors that affect behaviors of the people involved. If we overlook this perspective, we risk building theories, methods, processes, and models that do not account for vital factors, thus limiting their usefulness. As an example, human reluctance to change [17] may be more relevant to consider in software process improvement efforts than which procedures to update or which tools to introduce. Still, most research concerning software process improvement focuses on the change itself rather than the people who will be required to change their behavior [18, 19].

We acknowledge that the research area concerned with human aspects of software engineering is not unexplored. There have, for example, been special issue articles published in well-circulated journals within the software engineering field based on work at the Cooperative and Human Aspects of Software Engineering (CHASE) workshop series [20]. Two of the most intensively studied constructs thus far are motivation and personality [21–26]. The most recent review on motivation identified several factors that influence software engineers' motivation and highlighted that researchers have not thoroughly explored motivation in the context of agile practices [24]. Regarding personality, a review by Cruz, da Silva, and Capretz [26] concluded that research thus far has provided conflicting evidence, suggesting that personality research is immature and that direct application of the findings may not produce the desired effects.

There are, however, indications that software engineering researchers seldom use theories developed by the social sciences in their endeavors. For example, agile development emphasizes team aspects and collaboration and is, by its nature, associated with sociology and social psychology [27]. Still, a brief analysis of the publications included in four literature reviews related to agile development [28–31] revealed that none used sociology or social psychology theories. Social scientists from fields such as psychology and sociology have been studying human behavior for over a century [32]. We argue that the rich knowledge of human behavior found in these disciplines should be used more extensively to advance software engineering research, both directly, through the use of existing theories and concepts, and indirectly, through research

design and methodology. Without a broad and systematic consideration of social science results and methods, software engineering researchers risk having to "reinvent the wheel".

The primary objective of our research was therefore to *advance software engineering by more profoundly considering humans and their behavior*. We strove to both improve software development (i.e. the practice) and software engineering studies (i.e. the research area). Software development is composed of a multitude of diverse activities and streamlining these is beyond the scope of a single research discipline. Such challenges call for an interdisciplinary research approach that combines software engineering domain knowledge with the social sciences' in-depth understandings of human behavior. An inspiration is behavioral economics and the relative importance that this sub-field of economics has gained in a relatively short time span.

Considering the broad nature of our primary objective, we initially reviewed the scientific literature and explored industry needs to gain familiarity and acquire new insights into our area of research. Since the challenges in our field are in a preliminary stage, we argue that a general understanding of our research area may be beneficial. Research efforts without a distinct direction could, however, lead to vague and fragmented results that fail to provide rich and profound insights. Drawing on the knowledge achieved through our initial studies we, therefore, appended an additional, more confined, research objective directed toward exploring a specific phenomenon: organizational change.

The need to adapt and reinvent is crucial in software businesses, where rapid changes in influential technologies, the inherent flexibility of software, and continually evolving methodologies create an uncertain organizational environment [33, 34]. Implementing changes in organizations, however, has proven to be challenging, and many of these efforts fail to achieve their intended aims [35, 36]. Changes are a considerable source of stress for employees [37–39] as they add uncertainty to organizational life, which adversely affects employee well-being, organizational productivity, and product quality [40–42]. If software organizations are to maintain a high rate of change while simultaneously ensuring motivation and healthy stress levels among employees, they must manage these changes as smoothly and efficiently as possible. Therefore, our research's secondary objective was to *improve software companies' organizational changes by considering humans and their behavior*.

We conducted our studies at the intersection between software engineering and social science. We recognize that it is not evident whether our interdisciplinary efforts should be classified as software engineering research that utilizes methods, theories, and insights from social science or if we have conducted social science research in a software engineering context. Nevertheless, since our research objectives are sound, we argue that such faculty-related concerns are of secondary importance and that there are viable arguments for using the software engineering faculty as a placeholder for our efforts. Our research aligns with the IEEE definition [43] of software engineering research (i.e. the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software). Moreover, the stakeholders in our findings, those primarily interested in our results, are predominately software engineering practitioners.

1.1 Related work

There is a substantial body of knowledge related to the relatively broad research objectives of this thesis. In the following sections, we provide an overview of the scientific literature we deemed most relevant. The seven appended papers include more detailed descriptions of studies directly related to them.

Behavioral aspects of software engineering (Primary research objective)

In regard to our primary research objective, we identified (a) work and organizational psychology, (b) behavioral economics, (c) research considering behavioral aspects of software engineering, and (d) venues considering behavioral aspects of software engineering, as important to our work.

Work and organizational psychology

The American Psychology Association (APA) defines psychology as the study of the mind and behavior [44]. Work and organizational psychology (WOP), sometimes also referred to as industrial and organizational psychology, occupational psychology, or work psychology, is the application of psychology in the workplace and thus concerned with behavior at work [45].

Work and organizational psychology has existed for the last century and originated through the work of psychologists Hugo Münsterberg and Walter Dill Scott [46]. Early on, WOP research concentrated on physical work conditions such as lighting, ventilation, and noise level. From the beginning of the 1930s, however, interest in the social aspects of the work environment increased. In the 1950s, research oriented toward group and social level phenomena complemented the viewpoint related to individual differences that signified earlier WOP studies. New research topics arose and among the most critical directions were the sociotechnical and holistic perspective, the humanistic and motivational approach, and the cognitive view of decision processes [47].

Moreover, in the 1970s the issue of organizational culture was introduced into the WOP field by Pettigrew [48]. Since then, WOP researchers have increasingly adopted a multi-level approach, attempting to understand behavioral phenomena at the level of the organization, group and individual workers [49].

A number of challenges characterize the current state of the WOP field. There is a need to focus more on the organizational context, conduct studies that cut across the micro and the macro dimensions of behavior within organizations, and emphasize international aspects [47]. WOP research also raises essential questions about how to manage effectively in organizations given the increasing number of knowledge workers (for example software engineers) whose commitment is critical to organizational success [50, 51].

A sub-field to WOP that has recently received considerable attention from both researchers and practitioners is positive organizational psychology [52]. In contrast to mainstream WOP research, inquiries here are concerned with the study of positive subjective experiences (e.g. happiness, well-being, flow, and positive emotions) and positive traits (e.g. talents, interests, creativity, purpose, growth, and courage) in the workplace [52].

Behavioral economics

Behavioral economics is an interdisciplinary science that aims to establish descriptively accurate findings of human cognitive ability and social interaction with implications for economic behaviors and processes [53]. It uses models and knowledge from several neighboring sciences, but mostly from psychology [54]. Some scientists consider psychological economics a separate strand of behavioral economics that borrows solely from psychology, especially cognitive psychology [55]. Others single out behavioral finance, which argues that some financial phenomena can be better understood using models in which some agents are not fully rational [56].

We argue that the state of affairs in software engineering shares similarities with the state of affairs in the field of economics before the rise of behavioral economics as one of its sub-disciplines. While the prevailing assumption in economics during the 20th century was that human beings tend to make rational choices to maximize their economic output, behavioral economists draw on psychology and sociology to explain economic phenomena [57]. Their research is thus grounded on hypotheses based on empirical data showing how human beings think and behave. This fundamental shift in the underlying assumptions of economics has helped to create theories that can better account for the evident fact that humans show both irrational and altruistic behavior, self-sabotaging their progress [58].

The historical evolution of first acknowledging the importance of behavioral and psychological factors, then mostly ignoring them and then reintroducing them can be seen within the field of economics [57, 58] and, thus, may indicate a natural progression in the maturation of any research field. Nowadays, behavioral economics is a prosperous scientific field with a variety of conferences and journals. It has had a broad effect on scientific thinking in the area of economics, and its researchers have been recipients of two Nobel Memorial Prizes in Economic Sciences. Daniel Kahneman was awarded the prize in 2002 for his foundational work with Amos Tversky and gave his Nobel lecture on "Bounded Maps of Rationality: Psychology for Behavioral Economics" [59]. Richard Thaler was awarded the prize in 2017 for his contributions to behavioral economics. His research has shown how human traits systematically affect individual decisions and market outcomes [60].

Research considering behavioral aspects of software engineering

Researchers have repeatedly acknowledged the importance of behavioral aspects in software development [11–16]. There are literature reviews for several of the most studied behavioral constructs: motivation, personality, stress, intention to leave, organizational culture, organizational change, and organizational learning. Four literature reviews have covered motivation in software engineering [21–24]. Beecham, Baddoo, Hall, Robinson, and Sharp [21] stated that software engineers are likely to be motivated according to their characteristics (e.g. their need for variety) and their internal controls (e.g. their personality). Sharp, Baddoo, Beecham, Hall, and Robinson [22] have explored models used in motivation research. They concluded that motivation in software engineering is

poorly understood and that the models proposed in studies are fairly disparate and disjointed. Also, the most recent review on motivation supported this position [24]. The authors claimed that there is no clear understanding of what motivates software engineers, how they are motivated, or what benefits can be derived from motivating them.

Furthermore, Ghapanchi and Aurum reviewed intention to leave among information technology employees [61]. Their study, which included 72 publications from 1980 to 2008, identified 70 drivers and classified them into the five broad categories of individual, organizational, job-related, psychological, and environmental.

Regarding stress, a relatively small literature review by Maudgalya, Wallace, Daraiseh, and Salem [62], which included 12 publications, identified a link between burnout and the variables job task, role ambiguity, and role conflict.

Finally, a review by Cruz, da Silva, Fabio, and Capretz [26] concerning personality found that researchers have provided conflicting evidence, suggesting that personality research is immature, and that direct application of the findings may not produce the desired effects.

Venues considering behavioral aspects of software engineering

A number of workshops and conferences have addressed concerns related to the human aspects of software engineering over the years. Two examples are the CHASE workshops and psychology of programming (PoP) conference. The CHASE workshops have highlighted two main strands: the (a) human and (b) cooperative aspects of software engineering. Moreover, they have emphasized that SE activities typically occur in the context of a group or team [63]. Special issue articles based on work at CHASE have also been published in well-circulated journals within the software engineering field [20].

The original aims of the PoP workshops, which cover research in (a) computer programmers' cognition, (b) tools and methods for programming-related activities, and (c) programming education, were to help programmers work more efficiently and produce better software. Although the PoP interest group (PPIG), which hosts the PoP workshops, defines the term "programming" quite broadly to include any aspects of software development, the PoP workshop series emphasizes the individual perspective of programming. The research methods discussed and used in PoP most often have been adopted from cognitive psychology [64].

A number of conferences (e.g. Human-Centered Software Engineering (HCSE) and the Conference on Human Factors in Computing Systems (CHI)) have names that allude to our area of research. These, however, focus mainly on human-computer interaction or on usability. Similarly, we acknowledge that our research is related to the area of sociotechnical systems and that there have been several proposals for adapting such approaches to the design of software and information systems [65]. Although there are plenty of varied results and proposals in this area, they are generally more focused on the system to be delivered and the people that will use it than on the people that develop it.

Organizational change in software organizations (Secondary research objective)

Organizational change is defined both as the process by which an organization changes its structure, strategies, operational methods, technologies, or organizational culture to affect change within the organization and the effects of these changes on the organization [66]. There is a consensus among scientists and practitioners that the ability to manage organizational change is a core company competence and that the pace of change is increasing. Still, change processes remain challenging for organizations and many studies report a high failure rate for change initiatives [67–69].

Two approaches that have dominated the research on organizational change are planned change and emergent change [67, 70]. Planned change (also known as episodic or intermittent change), which originated with Kurt Lewin, led the field from the 1950s to the early 1980s. It is aimed at improving change efficiency through participation in change programs and assumes that change is centered around moving from one state to another in a structured manner; this has resulted in models that emphasize a structured set of steps. By contrast, emergent change approaches (also referred to as continuous or evolving change) see change as fluid, emerging, and continuous [69, 70]. Proponents of the emergent approaches which have dominated the field since the 1980s stress that in order to survive, organizations must develop the ability to change continuously [71].

Researchers have sought to identify factors that increase the likelihood of successfully implementing organizational changes. These efforts have revealed that one of the most critical factors for success is employee attitudes toward change [72, 73]. An organizational change cannot be considered successful without a change in employee behavior [74], which according to social psychology researchers [75] is controlled and predicted by attitudes. Several concepts for measuring various attitudes toward organizational change have been identified. According to Choi [76], the four most commonly used concepts are: readiness for change, openness to change, commitment to change, and cynicism about organizational change. These attitude concepts are susceptible to situational variables and may change over time as the individuals experience change; they are, therefore, better conceptualized as states than as personality traits.

For the past 15 years, change-related software engineering studies have often explored industry adoption of agile methods [29, 77, 78]. An area of particular interest has been to identify factors and challenges limiting agile transitions [79, 80]. The research suggests that management support, customizing agile methods, and proper guidelines are among the most critical factors. Moreover, several inquiries are cases or action research studies in which the researchers followed agile adoption processes at one or a few companies [81–85].

Another research area concerning change that has gained significant attention is Software Process Improvement (SPI) [86, 87]. SPI models such as the Capability Maturity Model (CMM) along with standards such as ISO's SPICE (ISO/IEC-15504 1998) focus on improving processes [87]. The SPI research is motivated by the assumption that there is a direct relationship between process quality and the quality of the produced software [88]. The aim of SPI is thus to increase product quality, but also to reduce time to market and production costs [88]. Researchers have suggested that SPI may not be delivering the benefits promised because insufficient attention has been paid to the human aspects of its implementation [18]. Furthermore, a review study by Lavall and Robillard [19] identified developers' resistance toward SPI as one of seven factors that have an impact on the implementation of SPI.

Moreover, there are a few software engineering studies that do not focus on any specific organizational change; instead, they analyze change in software companies more generally. Two studies have shown that organizational change has a momentary adverse effect on quality [89, 90]. Another study indicated that transformational leaders could act as change agents and, thereby, facilitate the change process [91].

Overall, studies concerning organizational change in software companies have, so far, focused on identifying and evaluating what processes and practices to improve (i.e. *what* to change), rather than identifying the best way in which actual improvements should be implemented in the organization (i.e. *how* to change). To the best of our knowledge, only one publication has focused on the human-oriented factors [92], exploring resistance to change.

1.2 Narrative and study objectives

In this section, we present an overview of the seven studies that are appended to this thesis, and provide a narrative explaining the underlying reasoning that justifies our research objectives. The studies we conducted are, to a certain extent, dependent on one another and aligned with our research objectives. However, since we conducted our research in close collaboration with software companies, our study choices were partially controlled by external events. For example, to collect relevant data, our organizational change case studies (Papers C to F), had to be synchronized with an actual change in one of our partner companies. Moreover, the result of one study motivated, and in some cases also provided input to, the subsequent studies.

Figure 1.1 presents an overview of our papers. As shown, the study objectives of the first two papers relate to our overall primary research objective: to improve software engineering by considering behavioral aspects. These papers thus contribute by improving the software engineering research discipline (i.e. how to conduct research). The other five papers relate to our secondary research objective (i.e. to improve organizational change efforts in software companies by considering behavioral aspects), contributing primarily by improving software development. The latter studies focused on two behavioral constructs: *attitudes toward organizational change* and *organizational values*.

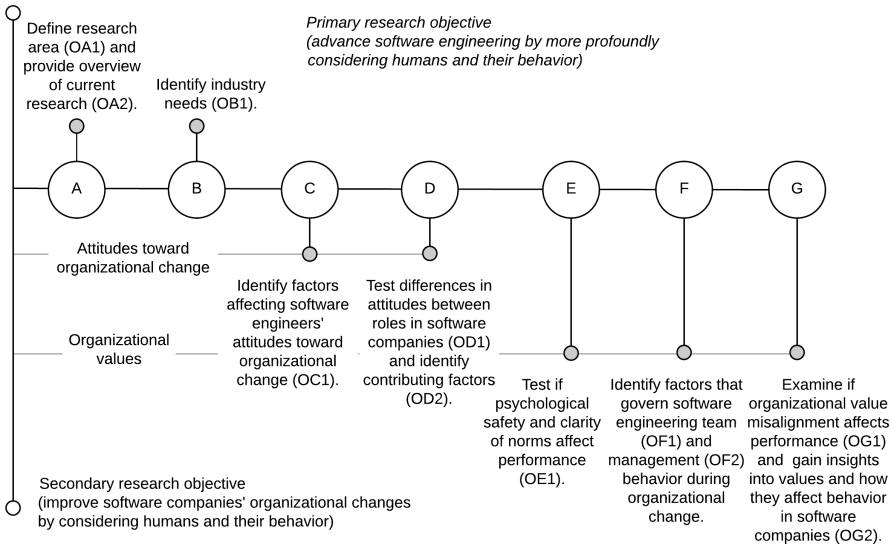

Figure 1.1: The figure presents the appended papers respective study objectives (O[*Paper*][*Objective*] in the figure) and their relation to the two overall research objectives.

(Paper A) Behavioral software engineering: A definition and systematic literature review

Rationale We sought to create a platform for further research by delineating the scope of our research and gaining insight into current research related to said scope.

Objectives The objectives of this paper were to define a research area concerned with psychological and social aspects of software engineering (OA1 in Figure 1.1) and provide a more detailed overview of the research conducted thus far (OA2). The paper contributed to the primary research objective by extending the knowledge of the research area concerned with behavioral aspects of software engineering.

(Paper B) Human factors related challenges in software engineering—An industrial perspective

Rationale The research overview that we presented in Paper A was rich and thorough. Nonetheless, it provided little information about the areas or activities that practitioners thought would gain the most from more in-depth knowledge into the human aspects. Therefore, to further expand our insights into behavioral software engineering and make an informed decision regarding future research, we consulted the software industry.

Objectives The objective was to identify areas for improvement in software organizations that require additional insights into human behaviors as part of the solution (OB1). The paper provided direction to future research, thus contributing to the primary research objective.

(Paper C) An initial analysis of software engineers' attitudes towards organizational change

Rationale To meet the software industry challenges identified in Paper B, we focused our efforts on organizational change. As a starting point in this research, we chose to examine change from individual software engineers' perspective by exploring their attitudes toward change. Attitudes had been previously identified as one of the most critical factors in the change process [72, 73].

Objectives The objective was to create, verify, and validate a model that predicted software engineers' attitudes toward organizational change (OC1). The paper contributed by extending the knowledge on organizational change in software companies (i.e. the secondary research objective).

(Paper D) An initial analysis of differences in software engineers' attitudes towards organizational change

Rationale To broaden and complement the knowledge gained through examining the individual perspective in Paper C, we explored attitudes toward change from an organizational perspective by testing differences between roles.

Objectives The objectives were to test differences in attitudes toward organizational change between roles in software organizations (OD1) and identify the factors that contribute to these presumed differences (OD2). Like the previous paper, this paper also added insights into organizational change efforts in software companies, thus contributing to the secondary research objective.

(Paper E) Psychological safety and norm clarity in software engineering teams

Rationale To comprehend the relationship between group norms and organizational change that we identified in Paper D, we required additional insights into norms and their effects on software teams. In particular, we wanted to know if efficient software teams enforced a specific set of group norms, or if teams could function well under various norms as long as they were known and clear.

Objectives The objective was to test if psychological safety (a specific set of social norms) and clarity of norms associate positively with software teams' performance (OE1). This paper indirectly contributed to the secondary research question by adding insights into software teams' social norms, which affect organizational change efforts.

(Paper F) The link between team behavior and organizational change

Rationale Paper D indicated that software engineering teams are affected by social influence and form their attitudes toward organizational change collectively. Therefore, if software companies are to maintain a high rate of change while simultaneously ensuring motivation and healthy stress levels among their employees, additional insights into the link between team behavior and organizational change are vital.

Objectives The overall purpose was to gain insights into the relationship between software engineering teams' behavior and organizational change. To address this aim, we defined two objectives that allowed us to acquire a broader and more profound understanding of the change process by portraying it from dual perspectives. Our first objective was to identify the factors, for example, norms, beliefs, feelings, and attitudes, which govern software engineering teams' attitudes and behaviors during organizational change (OF1). Our second objective was to identify the factors that govern management behavior and determine the reasoning and decisions of management during the change process (OF2). These objectives relate directly to our secondary research aim.

Moreover, this study also explored if and in what ways the activity-based working (ABW) environment (an office environment solution in which employees do not have specific seats or desks of their own) affects software engineering teamwork. This research is important since ABW is becoming increasingly popular among software organizations. It contributes to our primary research objective as an example that demonstrates the benefits of BSE inquiries.

(Paper G) Misaligned values in software engineering organizations

Rationale The findings in Paper D and Paper G indicated that differences in shared values between organizational groups adversely affected organizational change. In this final study, we sought more profound insights into such between-group value misalignment.

Objectives To examine if discrepancies in values between organizational groups affect software companies' performance (OG1). We also aimed to extend the knowledge of organizational values more broadly, using an exploratory research approach; accordingly, our secondary objective was to gain general insights into organizational values and how they affect behavior and performance in software companies (OG2). This study contributed to our secondary research objective by providing additional knowledge on between-group value misalignment, which affects organizational change efforts.

Paper	Size	Type	Process	Domain
C, D, and E	Large	Product development	Agile and plan driven	Safety-critical systems
B, F, and G	Large	Product development	Agile and plan driven	High-level systems and low-level components
B, E, and G	Large	Consultant	Agile and plan driven	High-level systems
G	Large	Product development	Scaled agile	Systems used by the automotive industry
B	Large	Product development	Scaled agile	Telecommunication software
E	Large	Product development	Scaled agile	Components for the power and automation industry
E and G	Medium	Consultant	Scaled agile	Communication systems and components
G	Medium	Product development	Agile and plan driven	Low-level software components
G	Medium	Product development	Agile	Low-level real-time components
B and E	Medium	Product development	Agile	Process simulation systems
B	Small	Product development	Agile	Computer vision components

Table 1.1: Overview of the participating companies. Large company: more than 5 000 employees, Small: less than 500 employees

1.3 Methods

The studies in this thesis were primarily conducted in Swedish industry. An overview of the participating companies is presented in Table 1.1, based on a set of characteristics that was acquired for all companies (size, company type, development process, and domain). As is shown, the eleven participating companies were active in a variety of domains, used different development procedures, varied in size, and developed both high-level systems and low-level components.

In Table 1.3, we present the research methods of the seven appended studies, which are detailed using five sub-features (research type, data type, collection method, analysis method, and the number of respondents) based on guidance from Easterbrook et al. [93]. The definitions of these sub-features, shown in Table 1.2, are not commonly accepted; still, we argue that they contribute by providing a general overview of our work.

These tables show that we have used a wide variety of methods, which adds value of its own by indicating that we have examined our research phenomenon from different perspectives using various vantage points. We have, for example, used variants of mixed method research designs [94] (sequential explanatory (Paper D), concurrent triangulation (Paper F), and sequential exploratory (Paper G)).

Sub-feature	Description
Data Type	Specifies how data were collected and is therefore only applicable to empirical research. Data are classified as quantitative, qualitative, or literature review. Quantitative data comprise anything that can be expressed as a number or quantified (ordinal, interval, or ratio scales) and lend themselves to statistical manipulation. Qualitative data, on the other hand, cannot be expressed as a number.
Collection Method	Defines what type of procedure that was used to collect the data, for example, interviews, focus groups, questionnaires, and observations.
Analysis Method	The data analysis method, for example, systematic literature review, thematic analysis, grounded theory, regression, analysis of variance, regression, and structural equation modeling.

Table 1.2: Description of sub-features.

We selected our approach based on the objectives for the respective study (see Table 1.3). Still, as can be seen, we have chosen to use qualitative methods in four out of the seven appended studies. Human behaviors, among individuals or in groups, are complex phenomena that sometimes cannot be sufficiently described or explained through statistics and other quantitative methods, and thus call for alternative approaches [95, 96]. Qualitative research methods are useful in a variety of situations. They are beneficial when addressing questions related to complex and versatile concepts such as behaviors, emotions, beliefs, and values. They are also favorable when identifying latent and hidden factors whose role in the phenomenon under investigation may not be apparent (e.g. social norms, gender roles, and religion) [97].

When using traditional quantitative data collection techniques (e.g. questionnaires), researchers have limited access to the reasoning behind respondents' answers. By contrast, qualitative techniques allow researchers to better explore underlying intrinsic processes. A key is open-ended questions, which provide participants with the opportunity to respond in their own words. Such answers are, unlike closed questions, not bound by researchers' knowledge but can stimulate responses that are meaningful and important to the participants [97]. Qualitative approaches are thus favorable when developing knowledge in poorly understood research areas such as ours, in which problems are in a preliminary stage [98].

Below, we briefly outline the data collection and analysis methods that we used and provide a short description of our rationale for the method chosen for each appended publication.

Publication	Data type	Collection method	Analysis method	No of respondents
Paper A	Quantitative	Literature review	SLR	250 publications
Paper B	Qualitative	Interview	Thematic	9
Paper C	Quantitative	Literature review and questionnaire	SLR and regression analysis	56 questionnaires
Paper D	Quantitative and qualitative	Questionnaire and interview	Analysis of variance and thematic analysis	51 questionnaires, 11 interviews
Paper E	Quantitative	Questionnaire	Regression	217
Paper F	Quantitative and qualitative	Focus group, interview, and questionnaire	Thematic and analysis of variance	42 focus groups, 9 interviews, 54 questionnaires
Paper G	Quantitative and qualitative	Interview and questionnaire	Thematic and correlation	14 interviews, 184 questionnaires

Table 1.3: Description of research methods per appended publication using five sub-features (presented in Table 1.2). SLR is short for systematic literature review.

Data collection methods

Interview There are three fundamental types of interviews: structured, unstructured, and semi-structured. In structured interviews, the interviewer asks a list of predetermined questions, with limited variation between interviews and no room for follow-up questions to responses that warrant further elaboration. Unstructured interviews, by contrast, do not reflect any preconceived theories or ideas. In semi-structured interviews, some questions are predetermined and asked of all candidates, while others arise spontaneously in a free-flowing conversation. According to Smith and Osborn [99], this procedure allows the interviewer and the interviewee to engage in a dialogue during which the scripted questions can be adapted to responses and where new subjects may be explored.

In our research, we wanted our interviews to be flexible but also to have a certain amount of guidance. On the one hand, we wanted the freedom that would allow us to explore new areas that might arise; on the other hand, we wished to maintain a structure that would permit replication of our results. We therefore chose to use semi-structured interviews in all of our qualitative studies (Paper B, D, F, and G). We used unstructured interviews (also called conversational interviews) for the manager interviews in Paper F. In that case, we deemed that the dialogue would require maximum flexibility for the research to adjust to situational circumstances.

Focus group Since the introduction of agile approaches, teams have become a dominant factor influencing organizational life. We used focus groups to gain insights into teams' collective behaviors during organizational change (Paper F). According to Kitzinger [100], such techniques are particularly useful for exploring groups' knowledge and experiences and examining not only what people think but also how they think and why they think in a particular way (i.e. their rationale [101]).

Questionnaire survey Survey research is a quantitative method whereby a researcher poses a set of predetermined questions to a sample of a population. Such an approach is especially useful when a researcher aims to describe the features of a large group [102] as it provides a general overview of the field being studied rather than a more in-depth analysis.

In our research, we used questionnaires in Papers C–G as an alternative to structured interviews since they can be distributed to a large sample at low cost. A potential downside of using questionnaires is that there are few incentives for the sample to answer the questionnaire, leading to a marked risk of low response rates (a response rate of over 60% is suggested for the survey to be considered valid [103]). In our appended studies, however, the response rate was over 80%. Also, we mainly used forced-choice questions with Likert scale response options [104], where the answers are on an ordinal scale, for example, from 'strongly agree' to 'strongly disagree'.

Data analysis methods

Literature review The procedure for the literature reviews (Paper A and Paper C) aimed to create a fair evaluation by using a trustworthy, rigorous, and auditable methodology. We based the reviews on the guidelines described by Kitchenham [105], which, in turn, were derived from practices in medical research and adjusted to suit software engineering.

When conducting the literature reviews, we used a predefined review protocol to reduce the possibility of researcher bias. The protocol described the review process, which included the following six stages: (1) analyzing the need for a systematic literature review, (2) selecting data sources, (3) selecting a search string, (4) defining research selection criteria, (5) defining the research selection process and (6) defining data extraction and synthesis.

Thematic analysis We analyzed all qualitative data based on positivist underpinnings [106] using an inductive approach (i.e. we drew theoretical and general conclusions from the data). To process the data, we used thematic analysis via the procedure developed by Braun and Clarke [107]. Thematic analysis is one of the most common forms of analysis in qualitative research [108]. It emphasizes pinpointing, examining, and recording themes within data. Themes are patterns across data sets that are important to the description of a phenomena and are associated with the study objectives [107, 109].

We recognize that there are other viable options besides thematic analysis that we could have used, such as grounded theory [110] or interpretative phenomenological analysis (IPA) [111]. However, since our aim was not to develop a novel theory to describe the findings (which is the primary aim of grounded theory [109]) and since IPA is not suited to capturing group-dynamic aspects [112], we deemed thematic analysis to be the most suitable alternative.

Analysis of variance We used analysis of variance, ANOVA (or the non-parametric equivalent), to determine whether there were any statistically significant differences between the means of three or more independent (unrelated) groups [113] in Paper D, F, and G. However, since ANOVA only identifies the existence of difference, Tukey's HSD post hoc test [114] was used in Paper D to determine which of the specific roles differed.

Prior to the analysis, we tested that the acquired data fulfilled the parametric statistical test assumptions. First, we inspected the histogram, box-plot and descriptive data for all constructs and verified that no outliers existed. Then we tested if the dependent variables were approximately normally distributed using the Shapiro-Wilk test of normality [115]. Finally, we tested the homogeneity of variance assumptions using Levene's test [116]. In Paper D, a few constructs violated homogeneity of variance and were, therefore, analyzed using the Welch ANOVA and the Games Howel post hoc test. Since the data in Paper F did not meet the normality assumption, we analyzed them using the Mann-Whitney U test [117].

Linear regression Furthermore, to determine the relationship between constructs, we used linear regression analysis in Paper C and Paper E. Before conducting the analysis, we tested statistical assumptions. Visual analyses of scatter plots for the variables indicated linear relationships. Further, we checked the homoscedasticity and normality of residuals with the Q-Q plot. Regarding autocorrelation, we calculated the Durbin-Watson values which were all between the two critical values of $1.5 < d < 2.5$. We could, therefore, assume that there was no first order linear autocorrelation in our multiple linear regression data. Finally, we determined the presence of multicollinearity. The variance inflation factors [118] were all well below three, indicating a small risk for multicollinearity.

The choice to use multiple linear regression analysis in Paper C and Paper E was based on the guidelines developed by Gefen et al. [119]. According to these guidelines, second generation data analysis techniques, such as partial least squares path analysis and LISREL, require a sample size at least 10 times the number of latent variables in the model. We recognize that this is just a rule of thumb and that there is no general consensus on the appropriate method for determining adequate sample size [120–122]. Nonetheless, with a sample size of 56, the choice of analysis method was by no means obvious but rather a border line case. We, therefore, acknowledge that using an alternative analysis method could have been a viable option that we will consider in future work. However, since our data passed the conditions for regression analysis we do not consider our choice as a major threat to the validity.

Methods in appended papers

(Paper A) Behavioral software engineering: A definition and systematic literature review

In our first study, we used theoretical reasoning to define a research area concerned with human factors of software engineering (OA1 in Figure 1.1). To create a common platform for our future research by identifying gaps and trends in the current research we used a literature review (OA1).

(Paper B) Human factors related challenges in software engineering– An industrial perspective

The purpose of this study was to identify the human factor related challenges that industry practitioners consider the most important to address (OB1). Since we aimed to gain a deeper understanding of the underlying constructs that influence software engineers' behavior, we chose a qualitative research method based on interviews. An alternative approach would have been to use a survey. Still, even if we recognized that a quantitative method would significantly increase the number of participants, we argued that lacked the knowledge needed to compile an appropriate survey and that the interviews would deepen our insights of the software engineers' reasoning and rational. Nonetheless, we acknowledge that it certainly would be valuable to test our results using, for example, a quantitative survey.

(Paper C) An initial analysis of software engineers' attitudes towards organizational change

We sought to compile a model that predicted software engineers' attitudes toward change (OC1). We achieved this by combining knowledge of the software engineering domain with existing organizational-psychology change theories. The models were verified using linear regression analyses of industry data collected through a survey. Instead of identifying the antecedents in the model using a literature review, we could have used a qualitative approach, for example, focus groups or individual interviews. Given that a great number of software engineering studies have identified factors affecting organizational change, in particular in the introduction of agile approaches, we chose to build our models on existing research rather than attempt to reinvent the wheel.

(Paper D) An initial analysis of differences in software engineers' attitudes towards organizational change

In this study, we aimed to test differences in attitudes toward organizational change between roles in a software engineering organization (OD1) and to explore the factors that contribute to these differences (OD2). We used a sequential explanatory mixed method in which collection and analysis of quantitative data is followed by collection and analysis of qualitative data [94]. The

purpose of such an approach was to use qualitative results to assist in explaining and interpreting the findings of a quantitative study. First, using the same data as in Paper C, we verified the presumed difference in attitudes between roles using analysis of variance. To gain in-depth insights into the factors that contribute to these differences, we then conducted semi-structured interviews which we processed using thematic analysis.

(Paper E) Psychological safety and norm clarity in software engineering teams

To examine if the well-known team constructs of psychological safety and team norm clarity associate positively with software developers' team performance and job satisfaction (OE1), we collected industry survey data which we analyzed using linear regression.

(Paper F) The link between team behavior and organizational change

In this study, we aimed to gain insights into the relationship between software engineering teams' behavior and organizational change (OF1) as well as explore whether and in what ways ABW environments affect teamwork (OF2). To achieve our two objectives, we used a triangulation approach, in which various methods were used to confirm, cross-validate, or corroborate findings within the study. For in-depth insights regarding the change process, we collected data from ten focus-group interviews with existing development teams and nine interviews with managers. To examine if the software engineers were affected by the introduction of ABW, we utilized a longitudinal survey.

(Paper G) Misaligned values in software engineering organizations

In the final study, we aimed to expand the body of knowledge on organizational values in software engineering (OG2) and, in particular, examine how discrepancies in values between organizational groups affect software companies' performance (OG1). To meet these objectives, we used a sequential exploratory method (i.e. the initial phase of qualitative research was followed by an quantitative inquiry) with the purpose of exploring the phenomenon [94]. First, we collected qualitative data by interviewing 14 employees working in four different software engineering organizations, aiming for a broad understanding of organizational values in software companies. The data were processed using thematic analysis. Then, to statistically test if value misalignment had affected the performance factors that we had identified in the qualitative analysis, we conducted a quantitative survey of seven organizations.

1.4 Results

In Table 1.4, we present an overview of the main results of our appended studies. Accordingly, it only includes the findings that we deem the most important in relation to the two overall research objectives. The results are grouped per study objective, outlined in Section 1.2. The following sections describe these findings in more detail.

Study objective	Result
OA1: Define research area.	We defined behavioral software engineering (BSE) as the study of cognitive, behavioral, and social aspects of software engineering performed by individuals, groups, or organizations.
OA2: To provide overview of current research.	• Few studies exist that consider multiple constructs from several units of analysis. • Few studies exist that included researchers from both the software engineering and social science faculties. • Researchers have, thus far, focused on a few BSE concepts in a limited number of areas.
OB1: Identify industry needs.	Four areas were identified: • Organizational change • Customer relations • One-dimensional solutions • Communication
OC1: Identify factors affecting attitudes toward change.	We identified three factors: • Knowledge of the intended change outcome • Understanding of the need for change • Sense of participation in the change process
OD1: Test differences in attitudes between organizational roles.	There were differences in attitudes toward change between organizational roles.
OD2: Identify factors that contribute to differences in attitudes toward change.	Software engineers evaluated change in relation to the norms of their peer group.
OE1: Test if psychological safety and clarity of norms affect performance.	Team performance and job satisfaction are predicted by both constructs. Team norm clarity is, however, a stronger predictor.
OF1: Identify factors that govern teams' behaviors during organizational change.	• Software companies' ability to conduct organizational change may be adversely affected by the introduction of agile approaches. • Software engineers formed their attitudes toward change according to their distinct professional identity.
OF2: Identify factors that govern management behavior during organizational change.	Complex organizational changes needed to be managed using an iterative approach in which each progression is evaluated based on empirical measurements to reduce confirmation bias.
OG1: Examine if organizational value misalignment affects performance.	Misaligned companies: • were less effective • were less satisfied • had lower levels of trust • had more conflicts.
OG2: Gain insights into values and how they affect behavior in software companies.	• If the meaning of the agile construct was not clarified and made common within the organization, an agile introduction could instead increase between-group value misalignment. • Companies can expect, and must take into account, that various organizational groups adopt new values at different paces, which at least temporarily increases between-group value misalignment. • A prerequisite for aligned values is an open dialogue.

Table 1.4: Overview of the main results for the appended papers. The study objectives are detailed in Section 1.2.

(Paper A) Behavioral software engineering: A definition and systematic literature review

The results of this study consisted of two main parts. In the first part, we presented a definition and motivation for a new research area that we named behavioral software engineering (BSE). We argue that the software engineering discipline would benefit from a clearly defined area of research concerned with realistic notions of human behavior that emphasizes different units of analysis in software development. A major inspiration for BSE is behavioral economics [54, 55] and the importance that this sub-field of the economics discipline has gained in recent years; consequently, we defined *behavioral software engineering as the study of cognitive, behavioral, and social aspects of software engineering performed by individuals, groups, or organizations.*

Work and organizational psychology researchers use three units of analysis (individual, group, and organization) to give structure to the activities in their studies [51]. Although software development is different from many other types of work, it is unlikely to constitute an entirely different type of human endeavor. Hence, we argue that these three aspects also constitute a structure that is relevant to BSE. The terms 'group' and 'organization' should here be considered in a general sense. The former includes different types of teams and other task-focused groups, while the latter includes more loose connections of multiple individuals such as communities.

In the second part, we reported the results of a systematic literature review that considers an extensive part of the BSE research area. In seeking information in books and publications, and by consulting experts in the fields of organizational psychology and social psychology, we identified 55 psychological concepts that we deemed relevant to the BSE research area and which we used to form the basis of the search strings in the literature review. On average, more than 500 papers were screened per BSE concept and after the screening and filtering steps had been completed, a total of 250 papers were finally included for further analysis.

The results showed that the BSE research area is growing and is considering an increasing number of concepts from psychology and social science. In addition, our review concluded that there are gaps in BSE research. Several concepts that are widely considered to be part of organizational and work psychology have not yet been studied in software engineering. We also identified a number of software engineering areas where no BSE research has been performed.

Furthermore, the results revealed that the research performed thus far has been unbalanced, with a heavy focus on a few BSE concepts in a limited number of software engineering areas and that few studies exist that consider multiple constructs from several units of analysis. The review also uncovered that less than 15% of the publications included authors from a social science faculty and that less than 5% of the publications included authors from both the software engineering and social sciences faculties.

(Paper B) Human factors related challenges in software engineering—An industrial perspective

This qualitative study indicated that (a) the interviewed practitioners believed that cognitive, behavioral, and social aspects should generally be considered more often in all software engineering activities and that (b) all roles would benefit from an increased knowledge of these human factors. In more detail, the thematic analysis of the interview transcripts resulted in the following four main descriptive themes.

Organizational change: The interview participants indicated that organizational changes are more frequent in software engineering companies compared to other businesses and that this affects both the software engineers' psychosocial health and their attitude toward change. Several participants also stated that organizational change efforts often failed because they tended to focus on only one aspect; they typically only recognized the organizational aspects, while the group and individual aspects were ignored. In addition, the participants identified a discrepancy in technical knowledge between the software engineers and the management and claimed that this discrepancy contributes to the fact that the changes efforts are often driven from the bottom up (i.e. initiated by the engineers).

Customer relations: The participants stated that software development is special in that it requires continuous cooperation between customer and supplier throughout the development process. One problem associated with customer relations was that software deliveries often introduced changes in the customer's organization. The participants thought that the customer employees felt threatened by these changes and that they, therefore, developed a negative attitude toward the supplier as a psychological defense. Also, the participants felt that the quality of the customer relationship is related to the contract type, where, for example, fixed-price contracts decrease parties' willingness to take responsibility and induce tension that can lead to conflicts between them in the long run.

One-dimensional solutions: The participants indicated that their companies sometimes had too narrow a focus and, as a consequence, tried to solve complex development related problems using overly simple solutions. An example of such a problem was cost and time estimation of development activities, where the companies sought solutions that only considered the individual developer's perspective and ignored organizational factors such as culture and politics. Also, theories regarding motivation and stress were also considered limited. Several interviewed managers claimed that motivation is fairly well-understood at the individual level, but that software companies need theories and practices that also incorporate group aspects.

Communication: Communication was mentioned both as an area of improvement, but also as an important psychological concept in itself. The participants saw verbal and written communication as central when creating a common understanding of their scope of work. Closely related to the scope of work is the requirement specification. To make the requirement specification less equivocal, the participants suggested that it should be examined from a psychological perspective.

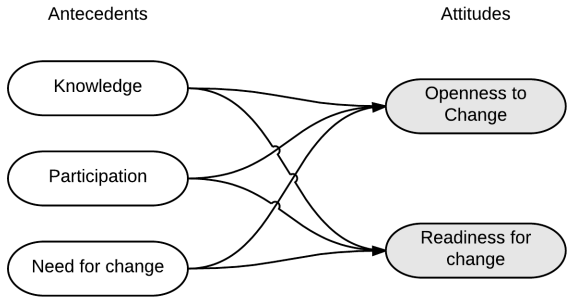

Figure 1.2: Model used to predict attitudes.

(Paper C) An initial analysis of software engineers' attitudes towards organizational change

We identified three underlying concepts with a significant impact on software engineers' attitudes toward organizational change: their knowledge of the intended change outcome, their understanding of the need for change, and their sense of participation in the change process. As the model in Figure 1.2 shows, we estimated *attitudes toward organizational change* using *openness to change* and *readiness for change*, two previously verified constructs. The explained variance of the *openness to change* regression model was high (44%) compared to other studies in social science, which adds support that our hypothesized model is a good first-order approximation and captures essential factors. Our results also provide some support for a hierarchy with respect to the three predictive concepts' degree of impact, where *knowledge* has a higher impact factor compared to *participation*, and *participation*, in turn, has a higher impact factor compared to *need for change*.

(Paper D) An initial analysis of differences in software engineers' attitudes towards organizational change

Our results confirmed that there were differences in attitudes toward organizational change between roles in a software engineering organization. The software engineers evaluated changes according to the norms of their peer group, meaning that a software engineer had a more positive attitude toward a specific change if the intended result facilitated the upholding of the group's norms. Also, the findings showed that the software engineers had more profound knowledge about team-based development (e.g. agile methods) compared to the line managers and that such discrepancies adversely affect the organization by reducing the trust.

(Paper E) Psychological safety and norm clarity in software engineering teams

The results demonstrated that team performance and job satisfaction are predicted by both psychological safety and team norm clarity. Clarity of team norms was a stronger predictor than psychological safety and it did not moderate the effects of psychological safety. This indicates that clarity of team norms is a significant concept on its own.

Also, our study highlighted the importance of adopting norms that the team members are aware of and understand. Holding distinct norms implies that team members recognize what behavior is accepted by the team, but it also suggests that team members know what behavior they can expect from their teammates. Such awareness arguably creates a predictable psychosocial work environment that reduces uncertainties.

(Paper F) The link between team behavior and organizational change

Our results suggested that software companies' ability to conduct organizational change is adversely affected by the introduction of agile approaches if these are not managed properly. Since the agile concept is elusive, its meaning is subject to interpretation, especially in regard to the organizational values (i.e. fundamental beliefs regarding what behaviors lead to organizational success) that signify agile organizations. According to our findings, this allows for different groups to create unique definitions, which become biased by each respective group's knowledge, norms, internal goals, and purposes. Consequently, there is an evident risk of groups adopting different organizational values. Such between-group value misalignment has significant adverse effects on subsequent change efforts.

Moreover, our findings indicated that software engineers form their attitudes toward change according to their distinct professional identity, which has undergone profound changes. Today, team membership, autonomy, and collective behavior are essential parts of software engineers' organizational character. Organizations must find ways to regulate the balance (i.e. control the relative strength) between organizational and professional identity to encourage altruistic team behavior and reduce in-group favoritism. Without such measures, efforts to scale agile change requiring inter-team collaboration are bound to fail.

Finally, the analysis suggested that practitioners must recognize that it is next to impossible to predict in advance the effects of substantial organizational changes on employees' behaviors, experiences, and attitudes. These changes thus call for an iterative change approach in which it is possible to make incremental improvements and evaluate each progression based on relevant empirical measurements to reduce the risk of confirmation bias.

(Paper G) Misaligned values in software engineering organizations

Our analysis showed that between-group value misalignment adversely affects organizational performance. The misaligned companies were less effective and less satisfied and had lower levels of trust and more conflicts. Several factors contribute to increased value misalignment. First, a prerequisite for aligned values is an open dialogue, without which values are bound to diverge. Our results suggested that shared values are seldom discussed in software organizations. Second, if the meaning of the agile construct is not clarified and made common within the organization, an agile introduction could instead increase between-group value misalignment. Third, it naturally takes time to change organizational values. Companies can expect, and must take into account, that various organizational groups adopt new values at different paces, which at least temporarily increases between-group value misalignment.

Moreover, our findings revealed that agile transition has not been simply a blessing and should not be considered a 'silver bullet'. As was reported by both the interviewed software engineers and the managers, the agile community has grown overly powerful and, at its worst, has created organizational values that prohibit questioning of the alleged superiority of agile methods.

1.5 Discussion

In outlining the behavioral software engineering (BSE) research area, reviewing contemporary research, and identifying industry needs, we created a scientific baseline that opens up avenues for further research in the field. Our results, for example, infer that BSE studies would benefit from becoming increasingly interdisciplinary and exploring challenges using multi-level inquiries that include individual, group, and organizational constructs.

Our findings also confirm the advantage of and need for BSE research by portraying organizational change in software companies from diverse perspectives, highlighting the importance of recognizing behavioral concerns such as software engineers' professional identity and misalignment of organizational values between groups.

Altogether, our research has provided encouraging, albeit initial results in the important interdisciplinary area of research concerned with behavioral aspects of software engineering.

1.5.1 Contributions

We present a summary of our main findings in Figure 1.3 grouped on the basis of which research objective they contribute to, namely those related to the software engineering discipline (i.e. the primary research objective) and those concerned with organizational change in software companies (i.e. the secondary research objective).

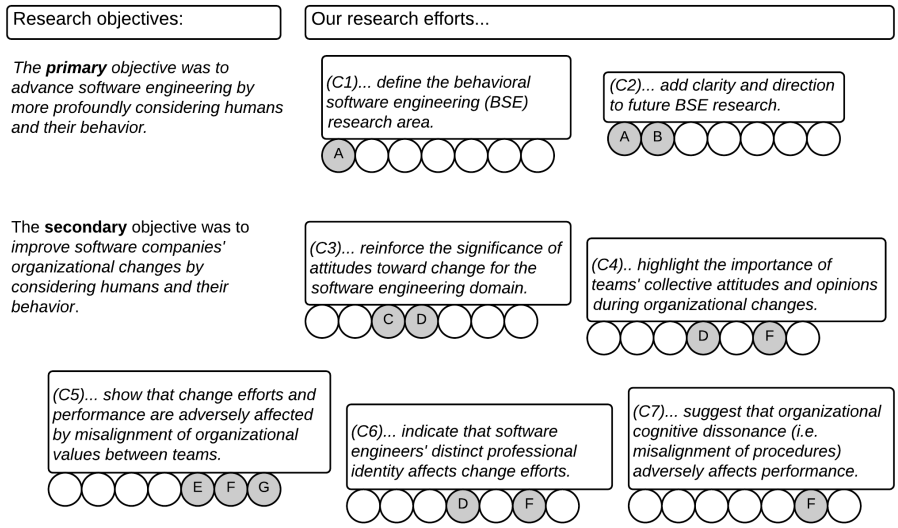

Figure 1.3: The figure presents a summary of the contributions grouped by research objective. It also shows appended papers that add support to the respective contribution.

Our collective organizational change studies (Papers C to G) also contribute to our primary research objective as a manifestation of BSE research. Firstly, these studies provided examples that confirm the benefits of utilizing existing social science constructs (e.g. attitudes toward change, alignment theory, norm clarity, and psychological safety) to the software engineering context. Secondly, this research showed that individual, group, and organizational related constructs (e.g. attitudes, group norms, and value misalignment) significantly affect organizational change efforts, thus reinforcing the notion in our BSE definition that three perspectives (i.e. individual, group, and organization) form a relevant starting point for giving structure to BSE research. Finally, our organizational change studies demonstrated the advantages of using qualitative methods in BSE research to create novel and in-depth insights into software engineering phenomena. Such insights facilitate the development of original, well-informed, and tailored solutions to software engineering related challenges.

Software engineering discipline

Our efforts contribute to the software engineering discipline by outlining an area of research concerned with realistic notions of human behavior that emphasizes the different units of analysis (i.e. individual, group, and organization) in software development (C1 in Figure 1.3). In the current scientific literature, research areas focusing on such concerns do exist; however, the overlap between these research areas and their respective coverage is not clear. While PPIG has focused on individual aspects and programming (i.e. one specific software engineering activity) it has directed less attention to group and organizational concerns and other software engineering activities. CHASE has

no distinct definition; it is instead outlined by eleven topics of interest, for example, 'Community-based development processes' and 'Distributed software development' [63].

We argue that having a clearly defined field of research adds value. In addition to its direct scientific value, we need the BSE definition for political reasons in order to legitimize and describe our research in dialogues both externally (i.e. with funding agencies and society at large) and internally (i.e. with software engineering researchers more focused on technical or process-related aspects). An inspiration for our research is behavioral economics and the importance that this sub-field of the economics discipline has gained in recent years [123]. We recognize, however, that our BSE definition is preliminary and not fixed. Even if we consider it to be a qualified starting-point, we appreciate that the definition will most certainly evolve as it is discussed further at conferences and workshops, and as the knowledge of the area is refined.

Moreover, our work strengthens the software engineering discipline by adding clarity and direction for future BSE research (C2 in Figure 1.3). Our review of the literature (Paper A) and analysis of industry needs (Paper B) together provide a baseline for researchers and confirm an industrial and academic need for a deeper understanding of the underlying mechanisms that govern software engineers' behavior. To the best of our knowledge, no previous review publications exist covering an extensive part of the BSE research area. Existing reviews have instead focused on individual constructs such as motivation [21,22] and personality [25].

Our review reveals that BSE researchers have been focused thus far on a few concepts, where the most frequently considered are communication, personality, and job satisfaction. It also shows that researchers from software engineering faculties conduct the vast majority of the studies. Our attitude study (Paper C) showed a likely consequence of this faculty homogeneity, which was that standard change models in work and organizational psychology (such as those suggested by Kotter [74]) were only used in 1 of the 42 included publications related to agile transition and organizational change.

Organizational change in software companies

A majority of our studies (five out of seven) explored various aspects of organizational change. The first two studies (Paper C and Paper D), examined attitudes toward organizational change, while our final three investigated the link between organizational values and organizational change.

The two attitude studies contribute mainly by applying a partially existing concept (i.e. attitudes toward organizational change) to a previously unexplored context (i.e. software engineering). Our studies reinforce the significance of attitude toward organizational change for the software engineering domain (C3 in Figure 1.3) and thereby conform to the large body of research postulating the concepts relevance [70, 73, 76, 124]. This is important because it opens up our field to the many studies and specific methods and interventions with which practitioners can improve organizational change in their organizations.

Moreover, our efforts also contribute by confirming differences in attitudes toward organizational change between roles in software companies, indicating that software engineers evaluate change in relation to the norms and values of their respective peer groups. Software engineers thus have positive attitudes toward change if the intended result would facilitate the upholding of their group norms. These findings are significant in that they highlight the importance of considering teams collective attitudes and opinions during organizational changes (C4 in Figure 1.3).

Our research also adds insights by indicating that software companies' ability to conduct organizational change is adversely affected by the introduction of agile approaches if these are not managed properly. Since the agile concept is elusive, its meaning is subject to interpretation, especially in regard to the organizational values that signify agile organizations. According to our findings, this allows for different groups to create unique definitions, which become biased by each respective group's prior knowledge, norms, internal goals, and purposes. Consequently, there is an evident risk of groups adopting different organizational values.

The conclusion that agile transitions risk increasing value differences has not been previously recognized. This finding is important since such between-group value misalignment adversely affects software organizations' potential to manage future organizational changes successfully (C5 in Figure 1.3). The software industry is currently transitioning to the use of scaled agile methods and additional insights into the factors that affect inter-team collaboration are thus crucial. Identifying existing barriers to organizational change and their underlying causes is a necessary initial step in improving the change process.

Our findings also highlight the importance of considering the congruence of values in software companies more generally. Current software engineering research on organizational values has focused on exploring the fit between specific values and agile approaches [79, 80, 125] rather than on alignment of values between teams. Our results show that between-group value misalignment has adverse effects on organizational performance.

Furthermore, our research contributes to the current literature by extending the knowledge of software engineering professional identity [126, 127]. The results suggest that software engineers form their attitudes toward change according to their distinct professional identity (C6 in Figure 1.3), which has undergone profound changes since the introduction of agile approaches. Today, team membership, autonomy, and collective behavior are essential parts of software engineers' organizational character.

According to previous research [128], it is the relative strength of professional identity and organizational identity that determines behaviors. Together with our findings, this implies that software engineers' professional identity is stronger than their organizational identity and that software engineers have become increasingly loyal to their teams at the expense of their organizational loyalty. Highly loyal teams tend to prioritize work that is beneficial to their respective teams over tasks that require inter-team collaboration [129,130]. Such behavior risks undermining conditions for scaled agile methods, in which effective inter-team collaboration is crucial.

Finally, our studies concerning organizational values suggest that in modern software organizations, teams have replaced individuals as the most critical entity. This is important and should reflect how software companies organize their internal processes. The shift from individuals to teams must not only affect methods directly related to the development of software; it must also affect other non-development related procedures such as performance reviews, wage discussions, and career paths. A lack of such company-wide alignment of procedures risks pulling the organization in two opposite directions thus creating an organizational cognitive dissonance that makes it challenging for teams to achieve their full potential (C7 in Figure 1.3).

1.5.2 Implications for practitioners

Our studies show that many challenges in software companies require interdisciplinary solutions, indicating that practitioners should more generally consider cognitive, behavioral, and social aspects of software engineering. Since such research is still rather uncommon, practitioners cannot solely rely on consulting software engineering literature. To find solutions, they must instead compile study results from several academic disciplines.

Moreover, we suggest that in order to avoid long-term adverse effects of between-group value misalignment, software companies must clarify the agile concept's conceptual meaning as applied to their organizations; a process known as 'tailoring' or 'contextualizing' [131, 132]. Such techniques are traditionally concerned with adjusting methods, processes, and practices, and seldom include activities delineating the behaviors, values, and beliefs that should form the basis for development. Nonetheless, without clarity on such significant matters, the agile concept will have different meanings in various parts of the organization.

We also recommend that software companies strengthen their organizational identities. This is particularly important for organizations in which inter-team collaboration is important. As a pre-step before implementing scaled agile methods, practitioners should regulate the balance between organizational and professional identities in order to encourage altruistic team behavior and reduce in-group favoritism. Without such preparatory measures, scaled agile change efforts are bound to fail.

Finally, we suggest that practitioners appreciate that it is next to impossible to predict in advance the effects of substantial organizational changes on employees' behaviors, experiences, and attitudes in environments as complex as software organizations. The outcome of a change effort is not solely dependent on the quality of the change, but also on the organization's current state, for example, its degree of between-group value misalignment. Organizational changes thus call for an iterative approach in which it is possible to make incremental improvements and evaluate each progression based on relevant empirical measurements to reduce the risk of confirmation bias.

1.5.3 Future work

Our research suggests that industry needs have significantly influenced BSE researchers' focus. Early on, when complexity of software systems was relatively low, software development required little collaboration since individual software engineers could take responsibility for complete deliveries. During this period, software companies strove to maximize the efforts of individual employees and BSE researchers tended to focus on individual constructs such as personality [26] and motivation [21]. When the complexity of systems increased, development required additional collaboration, and companies chose to organize their work in teams. This, in turn, influenced BSE researchers' efforts by intensifying studies related to group-level constructs such as team composition [133] and collaboration [134].

Nowadays, the complexity of software systems calls for extensive collaboration between multiple teams, making inter-team cooperation (scale agile methods) priority for software organizations. We, therefore, argue that future BSE research increasingly must emphasize organizational-level constructs such as organizational values, culture, and identity. However, we also expect that BSE studies would benefit from focusing on multiple units of analysis since the behavior of humans is too complex to be described using only one unit of analysis. Studies in work and organizational psychology support the need for such multi-level research [135, 136].

Moreover, our work on organizational change shows that researchers need additional insights into how norms and values are formed and maintained in software engineering groups, for example, agile development teams. Agile approaches emphasize the importance of autonomous and self-organizing development teams [137]; however, strengthening teams' autonomy may have undesirable consequences for software organizations. For example, if a team becomes detached from the organization, it risks developing a set of norms that is not aligned with that of other teams or with the organizational values. This may adversely affect inter-group collaboration. For an organization to maintain values that apply to the company as a whole, knowledge about how group norms arise and develop over time is required.

Our studies indicate that profound insights into software engineering professional identity and the underlying factors that form and govern it are long overdue. Several clues in our research indicated that software engineers' identity is changing. The engineers we interviewed strove for a different type of organization, questioned traditional hierarchical leadership, and were uncomfortable participating in political games that they felt took up too much energy while adding little of value to the organization's output. More profound insights into these matters would undoubtedly be beneficial for the software engineering field.

Furthermore, even if our findings suggest that between-group value misalignment and organizational performance are linked, we have reason to presume that the relationship is not simplistic or linear. For example, we expect that organizations with virtually no misalignment create static psychosocial environments with few or no disagreements that can stimulate creativity, improvement, and innovation. Drawing on the general agreement among scholars as to the relationship between tension levels and performance [138–140], we suggest that between-group value misalignment has an inverted U-shaped relation to performance. At low and high levels of value misalignment, organizations are less effective than at moderate levels. Still, such a hypothesis, of course, requires empirical support from future studies.

Finally, in future endeavors, we argue that BSE studies would benefit from becoming more interdisciplinary and securing a broad, serious, and systematic consideration of insights and methods from several disciplines. Including scientists from both social science and software engineering in the research teams could potentially achieve this.

1.6 Validity of research

It is of vital importance to know what constitutes sound research. Readers of software engineering publications (researchers and practitioners alike) must be sure that studies are trustworthy and provide robust results of real events [98, 141]. The value of research is thus dependent to a great extent on researchers' ability to demonstrate the credibility of their findings [142].

Researchers add credibility to their work by, for example, clearly describing their research methods, being reflective and outlining their biases, relating their results to previous research, and analyzing and describing threats to validity. The latter—threats to validity—is concerned with the relationship between conclusions and reality (i.e. how the conclusions might be wrong) [143]. The validity concept is not equivalent to quality since quality also includes other aspects such as relevance and replicability [144].

In the following sections, we first discuss the threats to the validity of our quantitative work and then of our qualitative inquiries. Finally, we provide an overall assessment of the validity of the appended studies drawing on the identified threats.

Validity threats for quantitative studies

Quantitative software engineering researchers commonly accept that they should describe how they have addressed issues related to construct validity, internal validity, external validity, and reliability [102, 144]. Below, we provide an overview of the threats using this classification. We recognize, however, that guidelines for validity, such as those provided by Runeson and Höst [102], are not intended as checklists for one-off post-hoc evaluations. With such approaches, researchers run the risk of overlooking severe threats to validity until it is too late to correct them. They should, instead, analyze validity and mitigate threats throughout the study process [144, 145].

Construct validity Construct validity reflects the extent to which the studied factors and constructs represent what the researchers had in mind and what the research objectives define [146]. Throughout our research, we aimed to utilize previously verified constructs rather than developing new scales and items in our questionnaires. Where no previous research existed, we were left with no alternatives but to operationalize the constructs ourselves (Paper C and Paper D). These studies have an elevated threat to construct validity.

Moreover, we acknowledge that one could question the use of self-assessment in our studies. In particular, the representativeness of the measurement *team performance* (in Paper E and Paper G) is a threat to validity. To raise the validity by triangulating such self-reported data, we collected project data in our final study (Paper G). We recognize, however, the questionable quality of that data. Objective measurement of team performance or rating by relevant managers, could, for example, help strengthen trust in the validity of the results.

Nevertheless, we deem that since we have analyzed the underlying items (using principal component analysis) and measured the internal consistency (using Chronbach's α), we can justify the use of the concepts and thus rate the threat to construct validity as acceptable.

Internal validity Internal validity concerns the validity of the examined causal relations in a study. We have strengthened the causal implications by building on previous work in which effects have been verified in another context. Also, in Papers D, F, and G, we triangulated the data by using multiple data sources.

Nonetheless, since we explored humans (as individuals or in groups), we were unable to control for all factors that alter behaviors. Additionally, we conducted all studies (except Paper A) in real-life contexts in which there are many complex interactions of factors affecting the phenomenon under inquiry. The threat to internal validity can, therefore, not be ignored.

External validity External validity describes the extent to which findings can be generalized to situations outside of the study context. An important variant of the external validity problem deals with selection bias, meaning that we conducted our research on a non-representative sample of the intended population. In our studies, the intended population was software engineers. We acknowledge that our sample has been biased and homogeneous, as over 90% of our participants were working for Swedish companies. That is a threat to our research's overall generalizability, and our conclusions must, therefore, be verified in other contextual settings and environments. Still, in three of our studies (Papers B, E, and G), we have collected data from multiple companies, adding some strength to the external validity.

Reliability Reliability refers to the ability to replicate the study with the same results. In a study by Open Science Collaboration, the researchers were unable to reproduce 36% of the original findings in 100 quantitative psychology studies [147]. We recognize this critique and have, in our studies, aimed to leverage reliability by presenting our methods in detail and using previously verified constructs in our questionnaires (applies to Papers E and G). All our studies were, however, industry case studies with unique contexts, making exact replicability challenging.

Even if we appreciate the 'replication crisis', we acknowledge that scholars are not unanimous regarding its existence [148] and also that some researchers emphasize triangulation, not just replication of studies [149].

Validity threats for qualitative studies

Among scholars, there is considerable debate over whether researchers should assess qualitative and quantitative methods according to the same criteria. Scientists generally seem to hold one of three opinions regarding standards of qualitative work. Some argue that it makes little sense to attempt to establish a set of generic criteria since there is no unified qualitative research paradigm [150]. Some suggest that, since different epistemological and ontological assumptions form the base for qualitative research, the established criteria for scientific rigor in quantitative research cannot apply to qualitative studies [151, 152]. Others claim that researchers should use the same broad criteria as quantitative research when assessing qualitative research [153–155].

We recognize these differences in opinion but acknowledge that they are part of a broader epistemological dispute regarding the nature of the knowledge produced by qualitative research [154]. Since we base our studies on positivist underpinnings (i.e. we assert, unlike social constructionists, that there is an underlying reality that can be studied [106]), we support the third position and maintain that both qualitative and quantitative research involves subjective perceptions that produce different perspectives [155–157]. As advised by Riege [158], we used the concepts confirmability, credibility, transferability, and dependability to assess validity threats, since they correspond to the traditional quantitative concepts of construct validity, internal validity, external validity, and reliability.

Furthermore, in our qualitative work, we ensured quality of the final two studies (Papers F and G) using the well-cited checklist *COREQ* [159]. The *COREQ* checklist, which is a structured consolidation of 22 qualitative checklists, was also used as a guide during these studies. When used in moderation, guidelines can help to guard against apparent errors and help to frame qualitative work as systematic and structured [160, 161]. Qualitative research does not, however, rely solely on detailed guidelines and practices to produce sound research, and an overly extreme methodological focus risks creating anxieties that hinder creativity and practice [162].

Confirmability Confirmability corresponds closely to construct validity and reflects whether the interpretation of data is drawn in a logical and unprejudiced manner [158]. To improve confirmability in our research, we used interview protocols which we pilot tested, defined our analysis processes in detail, and consulted experts from the social sciences during the analysis process.

Credibility Credibility is a parallel construct to internal validity and involves the approval of research findings [158]. To mitigate threats to credibility, we used triangulation techniques such as multiple data sources (Paper F and G), investigators (Paper B, D, F, and G), and methods (Paper D, F, and G). Also, we asked open-ended interview questions to ensure that participants could voice their opinion regarding any other issues that might affect our study topic (Paper B, D, F, and G).

Moreover, throughout the different stages of our most recent studies (Paper F and G), we attempted to foster reflexivity. Reflexivity has repeatedly been recognized as a crucial strategy in the process of generating knowledge through qualitative research [157, 163–165]. According to Berger [163], reflexivity is commonly viewed as the process of continuous internal dialogue and critical self-evaluation of a researcher's beliefs, as well as active acknowledgment and explicit recognition that these beliefs may affect the research process and outcome.

During the study processes, we regularly reflected on whether and how our preconceptions affected the outcome. Together we discussed the findings, proposed alternative interpretations, and acted as the devil's advocate [166,167]. Moreover, we recognize that we influenced the direction of our research and that we were a part of the very phenomena we studied merely by being involved with the participants. Our biases were therefore made plain at the outset of the reports to enhance the credibility of the result (Paper F and G) [157].

Transferability Transferability is analogous to the function of external validity or generalisation in traditional quantitative research [158]. It reflects the extent to which the findings are transferable to other settings [154].

To enhance transferability, we built on previous work and connected our results to similar studies in other contexts. We also provided detailed descriptions of the organizational contexts. The descriptions included a thorough account of our work so that readers can judge its potential for application to other times, places, people, and contexts. We note, however, that these descriptions are more profound in our recent papers (Papers F and G).

Dependability Dependability is comparable to the notion of reliability in quantitative research [158] and involves stability and consistency in the inquiry process. In our studies, aimed to increase dependability by presenting our methods in detail, using interview protocols, and by fostering reflexivity during the entire study process.

Validity assessment

As the previous sections emphasize, we have taken measures to mitigate threats to validity throughout the research process. Nonetheless, since we have studied human behavior in an industrial setting and since our empirical data were, in a few studies, limited, our results must be considered initial. They need verification from additional studies in other contextual settings using disparate research methods. In particular, we deem that our contributions related to attitudes (i.e. C3 and C4 presented in Figure 1.3 and detailed in Section 1.5.1) have an elevated threat to the validity since these are based on results from Paper C and D. In contrast, the threat to the validity of the contributions concerned with organizational values (C5, C6, and C7) is reduced since they are drawn from Papers E, F, and G.

1.7 Conclusions

Our research has provided encouraging, albeit initial results in the important interdisciplinary area of research concerned with behavioral aspects of software engineering. We have outlined directions for future research in defining the behavioral software engineering (BSE) research area, reviewing contemporary research, and identifying industry needs. Our results confirm the advantages of and need for BSE research by portraying organizational change in software companies from diverse perspectives, highlighting the significance of recognizing behavioral concerns such as attitudes toward organizational change, software engineers' professional identity, and value misalignment.

Drawing on our results, we suggest that software companies clarify the conceptual meaning of the agile concept as applied to their organizations in order to avoid long-term adverse effects of value misalignment. We also recommend that such companies, in which inter-team collaboration is critical, strengthen their organizational identities to counterbalance in-group favoritism and encourage altruistic team behavior.

In our research, we have frequently used qualitative approaches which generate findings with questionable generalizability, and our empirical data were in a few studies limited. The results must, therefore, be considered initial, and are in need of verification from additional studies in other contextual settings using disparate research methods.

The software industry is currently transitioning to the use of scaled agile methods, and additional insight into factors that facilitate inter-team collaboration is crucial. Besides replicating and verifying our results, we thus encourage BSE researchers to explore in-depth behavioral constructs related to the organizational level, such as organizational values and organizational identity.

Bibliography

[1] P. Lenberg, R. Feldt, and L. G. Wallgren, "Behavioral software engineering: A definition and systematic literature review," *Journal of Systems and Software*, vol. 107, pp. 15–37, 2015.

[2] P. Lenberg, R. Feldt, and L. G. Wallgren, "Human factors related challenges in software engineering – an industrial perspective," in *Proceedings of the 8th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2015*, pp. 43–49, IEEE, 2015.

[3] P. Lenberg, L. G. Wallgren Tengberg, and R. Feldt, "An initial analysis of software engineers' attitudes towards organizational change," *Empirical Software Engineering*, vol. 22, no. 4, pp. 2179–2205, 2017.

[4] P. Lenberg, E. Alegroth, R. Feldt, and L. G. Wallgren, "An initial analysis of differences in software engineers' attitudes towards organizational change," in *Proceedings of the 9th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2016*, pp. 1–7, IEEE, 2016.

[5] P. Lenberg and R. Feldt, "Psychological safety and norm clarity in software engineering teams," in *Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2018*, pp. 79–86, ACM, 2018.

[6] P. Lenberg, R. Feldt, and L. G. Wallgren Tengberg, "The link between team behavior and organizational change." *In submission to a journal*, 2018.

[7] P. Lenberg, R. Feldt, and L. G. Wallgren Tengberg, "Misaligned values in software engineering organizations." *Accepted for publication in Journal of Software: Evolution and Process*, 2018.

[8] P. Lenberg, R. Feldt, and L. G. Wallgren, "Towards a behavioral software engineering," in *Proceedings of the 7th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2014*, pp. 48–55, ACM, 2014.

[9] P. Lenberg, R. Feldt, L. G. W. Tengberg, I. Tidefors, and D. Graziotin, "Behavioral software engineering - guidelines for qualitative studies," *CoRR*, vol. abs/1712.08341, 2017.

[10] L. Gren and P. Lenberg, "The importance of conflict resolution techniques in autonomous agile teams," in *Proceedings of the First International Workshop on Autonomous Agile Teams, A-TEAMS'18*, 2018.

[11] G. M. Weinberg, *The psychology of computer programming*, vol. 932633420. Van Nostrand Reinhold New York, 1971.

[12] T. DeMarco and T. Lister, *Peopleware: Productive Projects and Teams*. Dorset House Publishing Company, Incorporated, 1999.

[13] A. Trendowicz and J. Münch, "Factors influencing software development productivitystate-of-the-art and industrial experiences," *Advances in computers*, vol. 77, pp. 185–241, 2009.

[14] O. Hazzan and I. Hadar, "Why and how can human-related measures support software development processes?," *Journal of Systems and Software*, vol. 81, no. 7, pp. 1248–1252, 2008.

[15] C. Amrit, M. Daneva, and D. Damian, "Human factors in software development: On its underlying theories and the value of learning from related disciplines. a guest editorial introduction to the special issue," *Information and Software Technology*, vol. 56, no. 12, pp. 1537–1542, 2014.

[16] F. Fagerholm, *Software Developer Experience: Case Studies in Lean-Agile and Open Source Environments*. PhD thesis, Helsingin yliopisto, 2015.

[17] S. Oreg, "Resistance to change: developing an individual differences measure.," *Journal of applied psychology*, vol. 88, no. 4, p. 680, 2003.

[18] N. Baddoo and T. Hall, "De-motivators for software process improvement: An analysis of practitioners' views," *Quality control and applied statistics*, vol. 49, no. 3, pp. 341–342, 2004.

[19] M. Lavallée and P. N. Robillard, "The impacts of software process improvement on developers: a systematic review," in *Proceedings of the 2012 International Conference on Software Engineering*, pp. 113–122, IEEE Press, 2012.

[20] C. R. de Souza, H. Sharp, J. Singer, L.-T. Cheng, and G. Venolia, "Guest editors' introduction: Cooperative and human aspects of software engineering," *Software, IEEE*, vol. 26, no. 6, pp. 17–19, 2009.

[21] S. Beecham, N. Baddoo, T. Hall, H. Robinson, and H. Sharp, "Motivation in software engineering: A systematic literature review," *Information and Software Technology*, vol. 50, no. 9, pp. 860–878, 2008.

[22] H. Sharp, N. Baddoo, S. Beecham, T. Hall, and H. Robinson, "Models of motivation in software engineering," *Information and Software Technology*, vol. 51, no. 1, pp. 219–233, 2009.

[23] T. Hall, N. Baddoo, S. Beecham, H. Robinson, and H. Sharp, "A systematic review of theory use in studies investigating the motivations of software engineers," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 18, no. 3, p. 10, 2009.

[24] A. C. C. França, T. B. Gouveia, P. C. Santos, C. A. Santana, and F. Q. da Silva, "Motivation in software engineering: A systematic review update," in *Evaluation & Assessment in Software Engineering (EASE 2011), 15th Annual Conference on*, pp. 154–163, IET, 2011.

[25] S. S. Cruz, F. Q. da Silva, C. V. Monteiro, P. Santos, and I. Rossilei, "Personality in software engineering: Preliminary findings from a systematic literature review," in *Evaluation & Assessment in Software Engineering (EASE 2011), 15th Annual Conference on*, pp. 1–10, IET, 2011.

[26] S. Cruz, F. Q. da Silva, and L. F. Capretz, "Forty years of research on personality in software engineering: A mapping study," *Computers in Human Behavior*, vol. 46, pp. 94–113, 2015.

[27] T. Dingsoyr, T. Dyba, and P. Abrahamsson, "A preliminary roadmap for empirical research on agile software development," in *Agile, 2008. AGILE'08. Conference*, pp. 83–94, IEEE, 2008.

[28] S. Jalali and C. Wohlin, "Global software engineering and agile practices: a systematic review," *Journal of Software: Evolution and Process*, vol. 24, no. 6, pp. 643–659, 2012.

[29] F. K. Chan and J. Y. Thong, "Acceptance of agile methodologies: A critical review and conceptual framework," *Decision Support Systems*, vol. 46, no. 4, pp. 803–814, 2009.

[30] S. V. Shrivastava *et al.*, "Distributed agile software development: A review," *arXiv preprint arXiv:1006.1955*, 2010.

[31] P. Sfetsos and I. Stamelos, "Empirical studies on quality in agile practices: A systematic literature review," in *Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the*, pp. 44–53, IEEE, 2010.

[32] F. D. Richard, C. F. Bond Jr, and J. J. Stokes-Zoota, "One hundred years of social psychology quantitatively described.," *Review of General Psychology*, vol. 7, no. 4, p. 331, 2003.

[33] J. Highsmith and A. Cockburn, "Agile software development: The business of innovation," *Computer*, vol. 34, no. 9, pp. 120–127, 2001.

[34] S. Nerur, R. Mahapatra, and G. Mangalaraj, "Challenges of migrating to agile methodologies," *Communications of the ACM*, vol. 48, no. 5, pp. 72–78, 2005.

[35] M. Beer, *Breaking the code of change*. Harvard Business Press, 2000.

[36] M. Meaney and C. Pung, "Mckinsey global results: Creating organizational transformations," *The McKinsey Quarterly, August*, pp. 1–7, 2008.

[37] J. E. Ferrie, M. J. Shipley, M. G. Marmot, S. Stansfeld, and G. D. Smith, "Health effects of anticipation of job change and non-employment: longitudinal data from the whitehall ii study," *Bmj*, vol. 311, no. 7015, pp. 1264–1269, 1995.

[38] C. A. Woodward, H. S. Shannon, C. Cunningham, J. McIntosh, B. Lendrum, D. Rosenbloom, and J. Brown, "The impact of re-engineering and other cost reduction strategies on the staff of a large teaching hospital: a longitudinal study," *Medical care*, vol. 37, no. 6, pp. 556–569, 1999.

[39] M. J. Schabracq and C. L. Cooper, "The changing nature of work and stress," *Journal of Managerial Psychology*, vol. 15, no. 3, pp. 227–241, 2000.

[40] A. Nelson, C. L. Cooper, and P. R. Jackson, "Uncertainty amidst change: The impact of privatization on employee job satisfaction and well-being," *Journal of Occupational and Organizational psychology*, vol. 68, no. 1, pp. 57–71, 1995.

[41] M. Vakola and I. Nikolaou, "Attitudes towards organizational change: What is the role of employees stress and commitment?," *Employee relations*, vol. 27, no. 2, pp. 160–174, 2005.

[42] A. E. Rafferty and M. A. Griffin, "Perceptions of organizational change: A stress and coping perspective.," *Journal of applied psychology*, vol. 91, no. 5, p. 1154, 2006.

[43] J. Radatz, A. Geraci, and F. Katki, "Ieee standard glossary of software engineering terminology," *IEEE Std*, vol. 610121990, no. 121990, p. 3, 1990.

[44] A. P. A. APA, "Apa dictionary of psychology." `https://dictionary.apa.org/psychology`. Accessed: 30-August-2018.

[45] P. M. Muchinsky, *Psychology applied to work: An introduction to industrial and organizational psychology* . Thomson Brooks/Cole Publishing Co, 1997.

[46] N. Chmiel, *An Introduction to Work and Organizational Psychology: An International Perspective*. John Wiley & Sons, 2017.

[47] L. W. Porter and B. Schneider, "What was, what is, and what may be in op/ob," *Annu. Rev. Organ. Psychol. Organ. Behav.*, vol. 1, no. 1, pp. 1–21, 2014.

[48] A. M. Pettigrew, "On studying organizational cultures," *Administrative science quarterly*, vol. 24, no. 4, pp. 570–581, 1979.

[49] S. W. Kozlowski, *The Oxford handbook of organizational psychology*, vol. 1. Oxford University Press, 2012.

[50] T. H. Davenport, *Thinking for a living: how to get better performances and results from knowledge workers*. Harvard Business Press, 2005.

[51] A. Furnham, *The psychology of behaviour at work*. Psychology Press, 2005.

[52] S. I. Donaldson and I. Ko, "Positive organizational psychology, behavior, and scholarship: A review of the emerging literature and evidence base," *The Journal of Positive Psychology*, vol. 5, no. 3, pp. 177–191, 2010.

[53] W. B. E. T. T. Do, "Behavioral economics: Past, present, future," *Advances in behavioral economics*, p. 1, 2011.

[54] M. Rabin, "A perspective on psychology and economics," *European economic review*, vol. 46, no. 4, pp. 657–685, 2002.

[55] J. F. Tomer, "What is behavioral economics?," *The Journal of Socio-Economics*, vol. 36, no. 3, pp. 463–479, 2007.

[56] N. Barberis and R. Thaler, "A survey of behavioral finance," *Handbook of the Economics of Finance*, vol. 1, pp. 1053–1128, 2003.

[57] L. Bruni and R. Sugden, "The road not taken: how psychology was removed from economics, and how it might be brought back*," *The Economic Journal*, vol. 117, no. 516, pp. 146–173, 2007.

[58] C. Lambert, "The marketplace of perceptions," *Harvard Magazine*, vol. 108, no. 4, p. 50, 2006.

[59] D. Kahneman, "Maps of bounded rationality: Psychology for behavioral economics," *American economic review*, vol. 93, no. 5, pp. 1449–1475, 2003.

[60] R. H. Thaler, "From cashews to nudges: The evolution of behavioral economics," *American Economic Review*, vol. 108, no. 6, pp. 1265–87, 2018.

[61] A. H. Ghapanchi and A. Aurum, "Antecedents to IT personnel's intentions to leave: A systematic literature review," *Journal of Systems and Software*, vol. 84, no. 2, pp. 238–249, 2011.

[62] T. Maudgalya, S. Wallace, N. Daraiseh, and S. Salem, "Workplace stress factors and burnoutamong information technology professionals: A systematic review," *Theoretical Issues in Ergonomics Science*, vol. 7, no. 3, pp. 285–297, 2006.

[63] R. Prikladnicki, Y. Dittrich, H. Sharp, C. De Souza, M. Cataldo, and R. Hoda, "Cooperative and human aspects of software engineering: Chase 2013," *SIGSOFT Softw. Eng. Notes*, vol. 38, pp. 34–37, Aug. 2013.

[64] J. Sajaniemi, "Psychology of programming: Looking into programmers' heads," *The Problems of Professionals*, p. 3, 2008.

[65] G. Baxter and I. Sommerville, "Socio-technical systems: From design methods to systems engineering," *Interacting with Computers*, vol. 23, no. 1, pp. 4–17, 2011.

[66] E. C. JUMA, *Organizational change as a strategy for competitive advantage at practical action Eastern Africa*. PhD thesis, SCHOOL OF BUSINESS, UNIVERSITY OF NAIROBI, 2014.

[67] B. Burnes, "Complexity theories and organizational change," *International Journal of Management Reviews*, vol. 7, no. 2, pp. 73–90, 2005.

[68] K. S. Whelan-Berry and K. A. Somerville, "Linking change drivers and the organizational change process: A review and synthesis," *Journal of Change Management*, vol. 10, no. 2, pp. 175–193, 2010.

[69] B. Burnes and P. Jackson, "Success and failure in organizational change: An exploration of the role of values," *Journal of Change Management*, vol. 11, no. 2, pp. 133–162, 2011.

[70] D. Bouckenooghe, "Positioning change recipients attitudes toward change in the organizational change literature," *The Journal of Applied Behavioral Science*, vol. 46, no. 4, pp. 500–531, 2010.

[71] S. L. Brown and K. M. Eisenhardt, "The art of continuous change: Linking complexity theory and time-paced evolution in relentlessly shifting organizations," *Administrative science quarterly*, pp. 1–34, 1997.

[72] A. E. Rafferty, N. L. Jimmieson, and A. A. Armenakis, "Change readiness a multilevel review," *Journal of Management*, vol. 39, pp. 110–135, 2013.

[73] S. Oreg, M. Vakola, and A. Armenakis, "Change recipients reactions to organizational change a 60-year review of quantitative studies," *The Journal of Applied Behavioral Science*, vol. 47, no. 4, pp. 461–524, 2011.

[74] J. P. Kotter and D. S. Cohen, *The heart of change: Real-life stories of how people change their organizations*. Harvard Business Press, 2002.

[75] I. Ajzen, "Nature and operation of attitudes," *Annual review of psychology*, vol. 52, no. 1, pp. 27–58, 2001.

[76] M. Choi, "Employees' attitudes toward organizational change: A literature review," *Human Resource Management*, vol. 50, no. 4, pp. 479–500, 2011.

[77] S. C. Misra, V. Kumar, and U. Kumar, "Identifying some important success factors in adopting agile software development practices," *Journal of Systems and Software*, vol. 82, no. 11, pp. 1869–1890, 2009.

[78] J. Iivari and N. Iivari, "The relationship between organizational culture and the deployment of agile methods," *Information and Software Technology*, vol. 53, no. 5, pp. 509–520, 2011.

[79] S. S. Hamid, M. H. N. M. Nasir, M. K. Othman, and R. Ahmadi, "Factors limiting the implementations of agile practices in the software industry: A pilot systematic review," *Indian Journal of Science and Technology*, vol. 8, no. 30, 2015.

[80] K. Dikert, M. Paasivaara, and C. Lassenius, "Challenges and success factors for large-scale agile transformations: A systematic literature review," *Journal of Systems and Software*, vol. 119, pp. 87–108, 2016.

[81] T. J. Gandomani, H. Zulzalil, A. A. A. Ghani, A. B. M. Sultan, and K. Y. Sharif, "Exploring key factors of pilot projects in agile transformation process using a grounded theory study," in *Information and Software Technologies*, pp. 146–158, Springer, 2013.

[82] R. Noordeloos, C. Manteli, and H. Van Vliet, "From rup to scrum in global software development: A case study," in *Global Software Engineering (ICGSE), 2012 IEEE Seventh International Conference on*, pp. 31–40, IEEE, 2012.

[83] S. Prokhorenko, "Skiing and boxing: Coaching product and enterprise teams," in *Agile Conference (AGILE), 2012*, pp. 191–196, IEEE, 2012.

[84] N. Nikitina, M. Kajko-Mattsson, and M. Stråle, "From scrum to scrumban: A case study of a process transition," in *Proceedings of the International Conference on Software and System Process*, pp. 140–149, IEEE Press, 2012.

[85] H. Hajjdiab, A. S. Taleb, and J. Ali, "An industrial case study for scrum adoption," *Journal of Software*, vol. 7, no. 1, pp. 237–242, 2012.

[86] F. J. Pino, F. García, and M. Piattini, "Software process improvement in small and medium software enterprises: a systematic review," *Software Quality Journal*, vol. 16, no. 2, pp. 237–261, 2008.

[87] M. Niazi, D. Wilson, and D. Zowghi, "Critical success factors for software process improvement implementation: an empirical study," *Software Process: Improvement and Practice*, vol. 11, no. 2, pp. 193–211, 2006.

[88] G. Cugola and C. Ghezzi, "Software processes: a retrospective and a path to the future," *Software Process: Improvement and Practice*, vol. 4, no. 3, pp. 101–123, 1998.

[89] S. Sato, H. Washizaki, Y. Fukazawa, S. Inoue, H. Ono, Y. Hanai, and M. Yamamoto, "Effects of organizational changes on product metrics and defects," in *Software Engineering Conference (APSEC, 2013 20th Asia-Pacific*, vol. 1, pp. 132–139, IEEE, 2013.

[90] A. Mockus, "Organizational volatility and its effects on software defects," in *Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering*, pp. 117–126, ACM, 2010.

[91] S. Angeline and S. Sudha, "Transformational leaders as change agents in information technology organisations," *International Journal of Applied Engineering Research*, vol. 10, no. 6, pp. 14299–14308, 2015.

[92] M. Serour and B. Henderson-Sellers, "Resistance to adoption of an oo software engineering process: An empirical study," in *University of Technology, Sydney, Australia. European and Mediterranean Conference on Information Systems (EMCIS)–2005*, 2005.

[93] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, "Selecting empirical methods for software engineering research," in *Guide to advanced empirical software engineering*, pp. 285–311, Springer, 2008.

[94] J. W. Creswell and J. D. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2017.

[95] C. B. Seaman, "Qualitative methods in empirical studies of software engineering," *IEEE Transactions on software engineering*, vol. 25, no. 4, pp. 557–572, 1999.

[96] A. Langley, "Strategies for theorizing from process data," *Academy of Management review*, vol. 24, no. 4, pp. 691–710, 1999.

[97] N. Mack, C. Woodsong, K. M. MacQueen, G. Guest, and E. Namey, *Qualitative research methods: a data collectors field guide.* Research Triangle Park North Carolina Family Health International [FHI] 2005., 2005.

[98] E. Fossey, C. Harvey, F. McDermott, and L. Davidson, "Understanding and evaluating qualitative research," *Australian and New Zealand journal of psychiatry*, vol. 36, no. 6, pp. 717–732, 2002.

[99] J. A. Smith, *Qualitative psychology: A practical guide to research methods.* Sage, 2007.

[100] J. Kitzinger, "Qualitative research. introducing focus groups.," *BMJ: British medical journal*, vol. 311, no. 7000, p. 299, 1995.

[101] D. L. Morgan, "Focus groups," *Annual review of sociology*, vol. 22, no. 1, pp. 129–152, 1996.

[102] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case study research in software engineering: Guidelines and examples.* John Wiley & Sons, 2012.

[103] E. Alégroth, *Visual GUI Testing: Automating High-level Software Testing in Industrial Practice.* Chalmers University of Technology, 2015.

[104] M. S. Matell and J. Jacoby, "Is there an optimal number of alternatives for likert-scale items? effects of testing time and scale properties.," *Journal of Applied Psychology*, vol. 56, no. 6, p. 506, 1972.

[105] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, p. 2004, 2004.

[106] J. Scotland, "Exploring the philosophical underpinnings of research: Relating ontology and epistemology to the methodology and methods of the scientific, interpretive, and critical research paradigms," *English Language Teaching*, vol. 5, no. 9, p. 9, 2012.

[107] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.

[108] C. Willig, *Introducing qualitative research in psychology.* McGraw-Hill Education (UK), 2013.

[109] N. K. Denzin and Y. S. Lincoln, *Handbook of qualitative research.* Sage publications, inc, 2005.

[110] B. G. Glaser and A. L. Strauss, *The discovery of grounded theory: Strategies for qualitative research.* Transaction publishers, 2009.

[111] J. A. Smith, "Reflecting on the development of interpretative phenomenological analysis and its contribution to qualitative research in psychology," *Qualitative research in psychology*, vol. 1, no. 1, pp. 39–54, 2004.

[112] L. Tomkins and V. Eatough, "Reflecting on the use of ipa with focus groups: Pitfalls and potentials," *Qualitative Research in Psychology*, vol. 7, no. 3, pp. 244–262, 2010.

[113] A. Agresti and M. Kateri, *Categorical data analysis*. Springer, 2011.

[114] J. W. Tukey, "Comparing individual means in the analysis of variance," *Biometrics*, pp. 99–114, 1949.

[115] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.

[116] H. Levene, "Robust tests for equality of variances," *Contributions to probability and statistics. Essays in honor of Harold Hotelling*, pp. 279–292, 1961.

[117] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947.

[118] R. M. Obrien, "A caution regarding rules of thumb for variance inflation factors," *Quality & Quantity*, vol. 41, no. 5, pp. 673–690, 2007.

[119] D. Gefen, D. Straub, and M.-C. Boudreau, "Structural equation modeling and regression: Guidelines for research practice," *Communications of the association for information systems*, vol. 4, no. 1, p. 7, 2000.

[120] J. C. Westland, "Lower bounds on sample size in structural equation modeling," *Electronic Commerce Research and Applications*, vol. 9, no. 6, pp. 476–487, 2010.

[121] J. C. Westland, *Structural Equation Models*. Springer International Publishing, 2006.

[122] E. J. Wolf, K. M. Harrington, S. L. Clark, and M. W. Miller, "Sample size requirements for structural equation models an evaluation of power, bias, and solution propriety," *Educational and psychological measurement*, vol. 73, no. 6, pp. 913–934, 2013.

[123] C. F. Camerer and G. Loewenstein, "Adam smith, behavioral economist," *The Journal of Economic Perspectives*, vol. 19, no. 3, pp. 131–145, 2005.

[124] S. K. Piderit, "Rethinking resistance and recognizing ambivalence: A multidimensional view of attitudes toward an organizational change," *Academy of management review*, vol. 25, no. 4, pp. 783–794, 2000.

[125] S. Chandra Misra, V. Kumar, and U. Kumar, "Identifying some critical changes required in adopting agile practices in traditional software development projects," *International Journal of Quality & Reliability Management*, vol. 27, no. 4, pp. 451–474, 2010.

[126] D. Jemielniak, "Software engineers or artists? programmers' identity choices," *Tamara Journal of Critical Organisation Inquiry*, vol. 7, no. 1/2, p. 21, 2008.

[127] A. Marks and D. Scholarios, "Revisiting technical workers: professional and organisational identities in the software industry," *New Technology, Work and Employment*, vol. 22, no. 2, pp. 98–117, 2007.

[128] D. R. Hekman, H. K. Steensma, G. A. Bigley, and J. F. Hereford, "Effects of organizational and professional identification on the relationship between administrators social influence and professional employees adoption of new work behavior.," *Journal of Applied Psychology*, vol. 94, no. 5, p. 1325, 2009.

[129] N. R. Branscombe, D. L. Wann, J. G. Noel, and J. Coleman, "In-group or out-group extemity: Importance of the threatened social identity," *Personality and Social Psychology Bulletin*, vol. 19, no. 4, pp. 381–388, 1993.

[130] M. Hewstone, M. Rubin, and H. Willis, "Intergroup bias," *Annual review of psychology*, vol. 53, no. 1, pp. 575–604, 2002.

[131] A. S. Campanelli and F. S. Parreiras, "Agile methods tailoring–a systematic literature review," *Journal of Systems and Software*, vol. 110, pp. 85–100, 2015.

[132] P. Kruchten, "Contextualizing agile software development," *Journal of Software: Evolution and Process*, vol. 25, no. 4, pp. 351–361, 2013.

[133] N. Gorla and Y. W. Lam, "Who should work with whom?: building effective software project teams," *Communications of the ACM*, vol. 47, no. 6, pp. 79–82, 2004.

[134] M. Hoegl and K. P. Parboteeah, "Creativity in innovative projects: How teamwork matters," *Journal of engineering and technology management*, vol. 24, no. 1-2, pp. 148–166, 2007.

[135] S. W. J. Kozlowski and B. S. Bell, "Work groups and teams in organizations," in *Work groups and teams in organizations*, vol. 12, pp. 412–469, Wiley, 2013.

[136] K. Crowston, "Processes as theory in information systems research," in *In Proceedings of the IFIP WG8.2 International Working Conference*, pp. 149–164, 2000.

[137] T. Chow and D.-B. Cao, "A survey study of critical success factors in agile software projects," *Journal of Systems and Software*, vol. 81, no. 6, pp. 961–971, 2008.

[138] C. K. De Dreu, "When too little or too much hurts: Evidence for a curvilinear relationship between task conflict and innovation in teams," *Journal of management*, vol. 32, no. 1, pp. 83–107, 2006.

[139] K. A. Jehn, "A multimethod examination of the benefits and detriments of intragroup conflict," *Administrative science quarterly*, pp. 256–282, 1995.

[140] E. Van de Vliert and C. de Dreu, "Optimising performance by stimulating conflict," *Int J Confl Manage*, vol. 5, pp. 211–22, 1994.

[141] J. Popay, A. Rogers, and G. Williams, "Rationale and standards for the systematic review of qualitative literature in health services research," *Qualitative health research*, vol. 8, no. 3, pp. 341–351, 1998.

[142] M. D. LeCompte and J. P. Goetz, "Problems of reliability and validity in ethnographic research," *Review of educational research*, vol. 52, no. 1, pp. 31–60, 1982.

[143] J. A. Maxwell, *Qualitative research design: An interactive approach*, vol. 41. Sage publications, 2012.

[144] R. Feldt and A. Magazinius, "Validity threats in empirical software engineering research-an initial survey.," in *SEKE*, pp. 374–379, 2010.

[145] L. Gren, "Standards of validity and the validity of standards in behavioral software engineering research: The perspective of psychological test theory," in *International Symposium on Empirical Software Engineering and Measurement (ESEM)*, (New York, NY, USA), pp. 374–379, ACM, 2018.

[146] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical software engineering*, vol. 14, no. 2, p. 131, 2009.

[147] O. S. Collaboration *et al.*, "Estimating the reproducibility of psychological science," *Science*, vol. 349, no. 6251, p. aac4716, 2015.

[148] D. T. Gilbert, G. King, S. Pettigrew, and T. D. Wilson, "Comment on estimating the reproducibility of psychological science," *Science*, vol. 351, no. 6277, pp. 1037–1037, 2016.

[149] M. Munafò and G. Davey Smith, "Robust research needs many lines of evidence," *Nature*, vol. 553, pp. 399–401, 1 2018.

[150] G. Rolfe, "Validity, trustworthiness and rigour: quality and the idea of qualitative research," *Journal of advanced nursing*, vol. 53, no. 3, pp. 304–310, 2006.

[151] Y. S. Lincoln and E. G. Guba, *Naturalistic inquiry*, vol. 75. Sage, 1985.

[152] A. Chapple and A. Rogers, "Explicit guidelines for qualitative research: a step in the right direction, a defence of the'soft'option, or a form of sociological imperialism?," *Family practice*, vol. 15, no. 6, pp. 556–561, 1998.

[153] K. Malterud, "Qualitative research: standards, challenges, and guidelines," *The lancet*, vol. 358, no. 9280, pp. 483–488, 2001.

[154] N. Mays and C. Pope, "Quality in qualitative health research," *Qualitative Research in Health Care, Third Edition*, pp. 82–101, 2007.

[155] C. Seale and D. Silverman, "Ensuring rigour in qualitative research," *The European Journal of Public Health*, vol. 7, no. 4, pp. 379–384, 1997.

[156] J. M. Morse, M. Barrett, M. Mayan, K. Olson, and J. Spiers, "Verification strategies for establishing reliability and validity in qualitative research," *International journal of qualitative methods*, vol. 1, no. 2, pp. 13–22, 2002.

[157] N. Mays and C. Pope, "Assessing quality in qualitative research," *Bmj*, vol. 320, no. 7226, pp. 50–52, 2000.

[158] A. M. Riege, "Validity and reliability tests in case study research: a literature review with hands-on applications for each research phase," *Qualitative market research: An international journal*, vol. 6, no. 2, pp. 75–86, 2003.

[159] A. Tong, P. Sainsbury, and J. Craig, "Consolidated criteria for reporting qualitative research (coreq): a 32-item checklist for interviews and focus groups," *International journal for quality in health care*, vol. 19, no. 6, pp. 349–357, 2007.

[160] M. LeGreco and S. J. Tracy, "Discourse tracing as qualitative practice," *Qualitative Inquiry*, 2009.

[161] S. J. Tracy, "Qualitative quality: Eight big-tent criteria for excellent qualitative research," *Qualitative inquiry*, vol. 16, no. 10, pp. 837–851, 2010.

[162] C. Seale, "Quality in qualitative research," *Qualitative inquiry*, vol. 5, no. 4, pp. 465–478, 1999.

[163] R. Berger, "Now i see it, now i dont: Researchers position and reflexivity in qualitative research," *Qualitative Research*, vol. 15, no. 2, pp. 219–234, 2015.

[164] H. DCruz, P. Gillingham, and S. Melendez, "Reflexivity, its meanings and relevance for social work: A critical review of the literature," *British Journal of Social Work*, vol. 37, no. 1, pp. 73–90, 2007.

[165] D. A. A. Ahmed, G. L. Hundt, and C. Blackburn, "Issues of gender, reflexivity and positionality in the field of disability: researching visual impairment in an arab society," *Qualitative Social Work*, vol. 10, no. 4, pp. 467–484, 2011.

[166] C. E. Hill, S. Knox, B. J. Thompson, E. N. Williams, S. A. Hess, and N. Ladany, "Consensual qualitative research: An update.," *Journal of counseling psychology*, vol. 52, no. 2, p. 196, 2005.

[167] C. E. Hill, B. J. Thompson, and E. N. Williams, "A guide to conducting consensual qualitative research," *The counseling psychologist*, vol. 25, no. 4, pp. 517–572, 1997.