

Overcoming Cultural Barriers to Being Agile in Distributed Teams

Darja Šmite^{1,2}, Nils Brede Moe^{1,2}, Javier Gonzalez-Huerta¹

¹ Blekinge Institute of Technology, Karlskrona, Sweden

² SINTEF ICT, Trondheim, Norway

{Darja.Smite | Nils.B.Moe | Javier.Gonzalez.Huerta}@bth.se

Abstract.

Context: Agile methods in offshored projects have become increasingly popular. Yet, many companies have found that the use of agile methods in coordination with companies located outside the regions of early agile adopters remains challenging. India has received particular attention as the leading destination of offshoring contracts due to significant cultural differences between sides of such contracts. Alarming differences are primarily rooted in the hierarchical business culture of Indian organizations and related command-and-control management behavior styles.

Objective: In this study, we attempt to understand whether cultural barriers persist in distributed projects in which Indian engineers work with a more empowering Swedish management, and if so, how to overcome them. The present work is an invited extension of a conference paper.

Method: We performed a multiple-case study in a mature agile company located in Sweden and a more hierarchical Indian vendor. We collected data from five group interviews with a total of 34 participants and five workshops with 96 participants in five distributed DevOps teams, including 36 Indian members, whose preferred behavior in different situations we surveyed.

Results: We identified twelve cultural barriers, six of which were classified as impediments to agile software development practices, and report on the manifestation of these barriers in five DevOps teams. Finally, we put forward recommendations to overcome the identified barriers and emphasize the importance of cultural training, especially when onboarding new team members.

Conclusions: Our findings confirm previously reported behaviors rooted in cultural differences that impede the adoption of agile approaches in offshore collaborations, and identify new barriers not previously reported. In contrast to the existing opinion that cultural characteristics are rigid and unchanging, we found that some barriers present at the beginning of the studied collaboration disappeared over time. Many offshore members reported behaving similarly to their onshore colleagues.

Keywords: Culture, Cultural Barriers, Agile, DevOps, Distributed Agile Teams.

1 Introduction

Software development projects designed by a single co-located agile team have become less common in recent years, and many agile software development environments have become geographically distributed. Software engineers then collaborate digitally with multiple sites of a single company or with subcontractors. Thus, agile teams within a project or program might commonly be spread over several time zones and geographic locations, implying that different national and organizational cultures are represented [1, 2]. Although new technology and processes have enabled better coordination and communication in globally distributed teams, such long-distance collaboration increases the complexity of software development operations, and various challenges emerge [3].

Such teams may face many obstacles due to geographical, temporal, and cultural variations. Problems have been shown to occur as a result of low availability of key personnel, absence of organizational support for unscheduled meetings between sites, and unbalanced or asymmetrical activity patterns from team members in virtual meetings [4]. Another example of such challenges arises in situations where more hierarchical business

organizations from certain Asian countries such as India or China collaborate with self-managing agile teams from Nordic countries. Agile development culture, characterized by flexibility and external focus, has been understood in contrast to hierarchical software development business culture emphasizing control and internal focus [5]. Further, self-managing teams are commonly given significant authority and responsibility for many aspects of their work, such as planning, scheduling, assigning tasks to members, and making decisions with economic consequences [6]. However, team members in organizations applying agile techniques in Asia have been reported by some studies as not asking questions when there was something they did not understand, nor engaging in discussions when they disagreed on how a problem might best be solved, nor participating in decision-making processes [1, 4, 7]. An interesting question is what happens when teams are set up with a combination of representatives from a mature agile organization and a more hierarchical organization (as illustrated in Fig. 1).

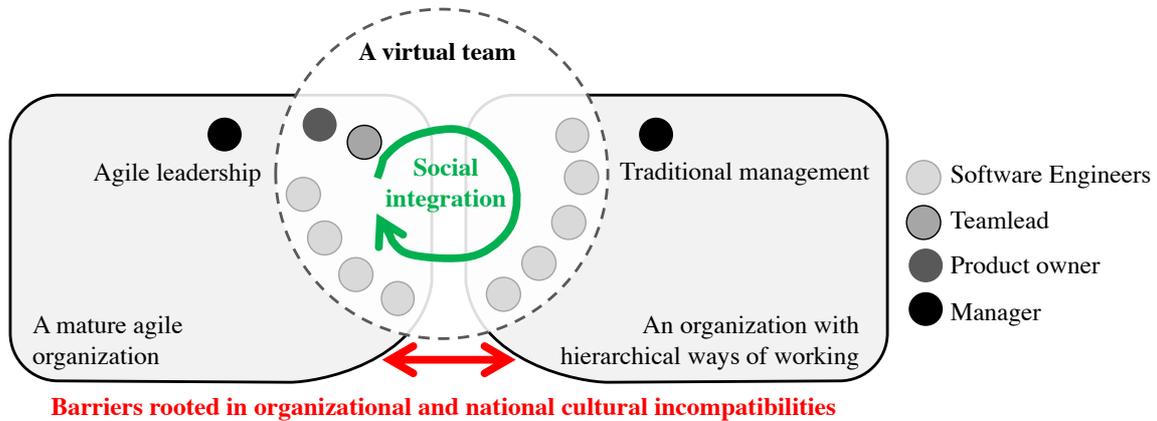


Fig. 1. DevOps team structure

In this study, we report our results from an empirical study of a Swedish company working with offshore engineers from an outsourcing vendor in India. The present work was invited as an extension of a prior work reporting preliminary findings on cultural barriers in distributed teams [8]. The extension is driven not only by the need to understand barriers to agility, but also focuses on how to succeed with the adoption of agile software development operations with globally distributed teams collaborating members from an Indian vendor, and, in particular, how to mitigate barriers rooted in organizational and national cultural incompatibilities via social integration. As a result, we explore specific cultural barriers, identifying behaviors impeding development agility, determining whether these behaviors prevail among offshore engineers working in distributed agile teams, and methods for companies and team members to foster beneficial social integration. In comparison with our earlier work [8], we have extended the related literature overview, added new empirical background information, new data, a new detailed discussion of how the culture studied changed significantly over time, and present the situations and behavioral differences contrary to agile software development methods that we identified in our research.

Our empirical study, therefore, addresses the following research questions.

RQ1: What are the barriers to agility in distributed teams with members from a hierarchical culture?

RQ2: What can agile teams do to overcome these barriers?

Our research contributes to the body of knowledge by providing information and analysis based on an empirical case study. The main contributions of this study include

- 1) validation of known cultural barriers and the discovery of cultural barriers that have not yet been discussed in related research,

- 2) detailed descriptions of manifestations of such cultural barriers in concrete situations related to agile practices and processes,
- 3) an empirical evaluation of the frequency of occurrence of cultural barriers in an industrial setting,
- 4) solutions to overcome cultural barriers in an agile software development context, and
- 5) evidence of the changes in varying national cultures and hierarchical business management models after exposure to a distinct cultural setting centered on enabling agile software development operations.

The remainder of this study is organized as follows. Section 2 summarizes the links between cultural differences and the behavior of Asian software engineers and the role of cultural differences observed in cases where companies have introduced agile software development approaches in offshore projects. Section 3 introduces our research methodology and the case company. The results of our study are presented in Section 4, followed by a discussion in Section 5. Section 6 concludes the study with a summary of the findings and implications for practice and further research.

2 Background and Related Work

Culture is related to and structures human lived experience in the world, beginning at birth, based on gestures, words, tones of voice, sounds, colors, smells, and body contact we experience [8]. A shared culture may be described as a collection of experiences and ways of relating in society that are familiar, recognizable, habitual, that “go without saying”, establishing a definition of “normal”. However, culture is complex and multifaceted concept; it can be attributed to a nation, an organization, a group, or even to an individual because it is shaped by human social environments [9]. Therefore, culture is a sensitive subject and not a depiction of rigid stereotypes [10]. However, common characteristics may exist that distinguish one culture from another [10]. In the present work, we focus on the impact of national and organizational culture on software development operations.

National cultures and cultural differences have been studied in depth by numerous social scientists (e.g., [9] and [11], among many others). These studies resulted in the identification of several overlapping cultural characteristics that were found to be common to representatives of various nations. National culture may determine or strongly affect preferred leadership styles and decision-making processes, perceptions of authorities, attitudes toward time, need for formalization, preferred communication and interaction styles, business etiquette, and motivation tools [9]. Similar to organizational incompatibilities with various operational methods [5, 12], incompatibilities in the national backgrounds and differences in ways of working have proven problematic in prior research [3, 13]. In fact, the larger the degree of difference between organizations and national cultures, the larger the cultural distance between the parties involved [14]. Below, we first explain the characteristics of agile software development operations and describe characteristics of organizational culture conducive to successful adoption of agile methodology. Then, we summarize research studies related to challenges in introducing agile ways of working in Asia, which are relevant to our empirical study.

2.1 Agile Ways of Working and Organizational Culture

Several researchers have emphasized the importance of cultural compatibility or fit between organizational cultures and the software development method they implement [5, 12], and companies are likely to encounter difficulties as a result of cultural incompatibilities. In particular, several studies have investigated the relationship between organizational culture and the use of agile methods [7, 15-17]. Agile ways of working stem from a group of methods united by a common philosophy, values, and principles that put certain requirements on teams as well as organizations.

Agile methods focus on teamwork and rely heavily on the ability of a software team to self-organize or self-manage [7, 18, 19]. Based on two large surveys of agile teams, Williams captures practices essential for teams to

be considered agile, being related to their ability to satisfy customers through early, continuous, and frequent delivery of valuable, working software. Among several prerequisites for this approach is staffing projects with motivated individuals who are given the required resources and authority to perform their job functions [20]. The principles of self-management and autonomy, central to agile ways of working, put certain demands on organizational culture, team composition, and behavioral norms [7, 21, 22]. Agile teams rely on people and their creativity rather than on processes, and emphasize mutual adjustment as the primary coordinating mechanism [23]. Agile teams consist of “a small number of people with complementary skills who are committed to a common purpose, set of performance goals, and approach for which they hold themselves mutually accountable.” [24]. By nature, self-managing teams are arenas for decision-making, problem-solving, and reflection, all facilitating knowledge creation. It has been shown that members of teams given authority beyond mere execution of tasks, for example, by mandates to make decisions about how tasks are done and by whom [25], are also more likely actively to question the assumptions behind their tasks. Such questioning of assumptions is an important basis for learning [26], which is needed to handle the complexity of software development. It is well-understood that ‘thinking outside the box’ enables teams and team members to integrate different streams of insights and propose solutions to the ever-growing complexity of software projects. While learning is the key in self-managing teams, such teams also need to allow operating norms and rules to change [27]. Further, Morgan emphasizes the importance of teams’ ability to engage in driving continuous improvement, and their ability to act on minimum critical specifications. With regard to necessary cultural traits, van Solingen et al. [28] argued that openness and an ability or willingness to discuss underlying problems and identify root issues are prerequisites for improvement and learning.

Team leadership function distributed among several team members has been considered a prerequisite for self-managing agile teams [19]. In general, it is understood that team members should share the authority to make decisions, rather than having one person (e.g., a team leader) that makes all significant decisions and judgements. Thus, while the traditional perspective of organizations oriented towards a single leader suggests that the leadership function is a specialized role, in self-managing agile teams, leadership is shared. Trust is a prerequisite for shared leadership in agile teams [19]. Without sufficient trust, team members will expend time and energy protecting, checking, and inspecting one another, as opposed to collaborating to provide value-added ideas.

Even though the need for external leaders is reduced in agile teams, external leadership remains important in agile software development for several reasons [29]. First, self-managing teams rarely delegate full decision-making authority [30]. It is often left to the leader to make key decisions regarding a team’s project. Second, external leaders are ideally suited to perform some kinds of activities, such as encouraging the team, managing the team’s boundaries [31], and dealing with unexpected problems or events that can occur. This means that the interaction between agile teams and leadership roles is distinct from traditional management, implying a coaching function [32]. Further, different types of team coaching exist, distinguishing between forms that are more supportive and reinforcing, and those that focus on identifying team problems and actively engaging in tasks. Finally, the external leader plays a key role in successful agile adoption. Therefore, leaders’ experiences from non-agile organizations (what can be referred to as their “baggage”) should be minimal [33], as such experience may be expected to slow the adoption of agile methods processes.

Based on a multi-case study of nine projects, Strode et al. [17] found that specific organizational cultural factors correlate with the effective use of an agile method. Their findings suggest that an organization is more likely to be successful if the following conditions are met. (1) The organization values feedback and learning. (2) Social interaction in the organization is trustful, collaborative, and competent. (3) A project manager acts as a facilitator, and the overall project management style is defined by collaborative leadership. (4) The organization values teamwork and is flexible and participative, encouraging social interaction. (5) The organization enables empowerment of developers within a project. (6) The organization is results-oriented. (7) Leadership in the organization is entrepreneurial, innovative, and risk-taking. (8) The organization is based on loyalty, mutual trust, and commitment [17]. Similar findings emerged from a study of 58 agile practitioners from 23 organizations in New Zealand and India [7]. Hoda et al. found that the support of senior management was a prerequisite for self-organizing agile

teams to be established and flourish, in terms of providing freedom and establishing an organizational culture of trust, echoing the results of a literature review performed by Dikert et al. [33].

There seems to be no consensus on whether agile working methods can be successfully implemented in different types of organizations. For example, Hoda et al. suggested that organizations with strict hierarchical structures are not conducive to self-organizing agile teams, because the hierarchy enforces a lack of openness marked by restricted and indirect lines of communication and feedback, which in turn leads to work environments based on motivation of employees by fear [7]. Similarly, Iivari et al. argued that if agile methods were adopted in an organization with a relatively strong hierarchical culture, agile methods may be expected to be formalized, which will make the combined model heavier and, as a consequence, such methods may start to lose some of their emergent agility [16]. In contrast, Dikert et al. found in their literature review that customizing the agile approach to organizations and organizational culture is one important factor affecting project success [33]. Similarly, Kautz et al. [15] found that agile development thrives in different organizational cultures, even in cultures with strong elements of control, as long as the four core values are present to a significant extent, including valuing *individuals and interactions* over processes and tools, *working software* over comprehensive documentation, *customer collaboration* over contract negotiation, and *responding to change* over following a plan). Furthermore, they argued that while organizational culture has an impact on implementations of agile development, in practice, often the method is adjusted to organizations.

2.2 Changing Agile Practices and Norms Over Time

The relationship between an organizational culture and agile ways of working is dynamic and therefore continuously evolving [16]. Further, in agile projects, agile practices change over time, as involved actors need to solve new problems implicating previously unknown interdependencies. New coordination mechanisms are gradually established through existing social coordination practices. This means that time perspective is significant and that studies on the compatibility between cultures and agile software development models should consider the dynamic nature of this relationship. In their study of two Malaysian and two Norwegian teams, Stray et al. [22] found that agile practices changed over time, for example, due to changing levels of authority, and that the retrospective meeting was a key event initiating the change.

Another reason for looking at the organizational culture from a time perspective is that personnel assignments change, as well. When adding new people to an already established agile team, the behavior of the new members might differ, and therefore, a given team might be forced to return to the subject of cultural differences and socialization repeatedly during its existence. It is thus essential to support new team members in adapting to an existing team's culture and work environment, which is especially difficult in virtual setups [34].

2.3 Agile Adoption in Asian Countries

Because national culture has been said to have a significant influence on organizational culture [9] and organizational culture may impact the use and success of agile ways of working [15, 17], the topic of understanding the use of agile methods and practices in companies located outside the locations of early adopters of agile methods, such as the US, Canada, and Western Europe has attracted considerable interest (most research on agile methods originates from these regions [35]). Agile adoption has been slower in Asian regions in comparison, which was confirmed by a study on DevOps [36] showing a steady development trend of DevOps in China, with a significant gap remaining compared with the international level.

While success stories on agile adoption exist, researchers and practitioners have wondered about the abilities of the companies and engineers from the Asian region, the primary recipients of offshoring contracts, to adopt agile software development operations methods, which are considered to be so distinct from their national culture. A number of studies have sought evidence of the successful use of agile methods in offshored projects [2, 3]. Kajko-Mattsson et al. [37] in a setup with vendors in India and China found that cultural problems were built on

strong and deep-rooted differences in mentality and culture. Aspects such as hierarchical behavior patterns or indirect or ambiguous communication were said to persist over several generations of programmers before being solved [37]. However, they were partially remedied via (1) team building exercises to blur the distinctions between different role levels, (2) imposition of collective task ownership, and (3) introduction of a neutral third-party agile coach. Further, Dorairaj et al. conducted a longitudinal study involving 55 participants from 38 different software companies in the USA, India, and Australia, and reported that senior management was required to provide distributed teams with significant support in terms of organizational culture, human resource management, financial sponsorship, infrastructure and technology, and customer liaison in order for the team members to work together as a single team despite cultural differences [38]. Stray et al. [22] studied Malaysian and Norwegian teams and found that the Malaysian developers had a very good implementation on one of the core agile practices as compared to the Norwegians – the daily standup meeting. Further, the Malaysian teams facilitated self-management and shared leadership by rotating the role of the Scrum Master. In their study on challenges in agile teams distributed over Japan and China, Ozawa et al. [39] explained that challenges were related to how culture influences the level of uncertainty accepted in agile project development operations. They found that by constructing user stories, team members with very vague communication styles were forced to ask the product owner for clarification. Vague user stories initiated a much closer interaction between Japanese and Chinese members and enabled two cultural groups to reduce tension, awkwardness, and unfamiliarity. Communication became much more frequent, and defects were detected and fixed earlier. However, some challenges could not be overcome, which was attributed to deep-rooted cultural characteristics.

Some infer successful adoption from a large number of practices reported as being followed. However, the validity of such studies and the associated research approach are questionable because a high level of commitment to the use of agile practices can be explained by the readiness to accept established rules in cultures with relatively hierarchical business structures based on high power distances between management and developers.

Other research studies have sought to improve the understanding of specific factors impeding the adoption of agile methods of working in Asian cultures [7, 13, 40-42] and successful approaches to applying them [2, 7]. Sundararajan et al. [43], in their case study of a US customer and Indian vendors, found that the very method used to implement agility – a hybrid model – was problematic. Sequential processes were planned at the beginning and end of the project, with agile processes sandwiched between [43]. In this way, they blended agile structures with the company's established hierarchical structure. From the research by Iivari [16] we know that such a setup can be expected to hinder an agile transformation, despite significant changes over time.

In Table 1, we summarize a list of concrete impeding behaviors rooted in national culture reported on managerial and engineering levels in related studies. The cited studies cover different countries within the Asian region, including India [7, 41, 42, 44], [37], China [37], Malaysia, and Singapore [40] as well as the Asia-Pacific region in general [42], and are either based on interviews or directly recorded experiences. A closer look at the behaviors of engineers in India and neighboring countries reveals that most, if not all, impeding behaviors were likely to have been caused by the hierarchical culture of the organizations studied and related management behavior, as suggested in related research [7, 17]. For example, Ayed et al. [40] reported that Malaysian and Singaporean engineers lacked the freedom to decide on their ways of working and, therefore, did not see any benefit to be realized from self-directed learning. We considered the question of whether a hierarchical business culture based on command and control, highlighted in numerous studies as being poisonous to agile methods [3, 13, 40-42], would allow its team members to shift their style of software development practices towards a more open, decentralized model under the influence of a more empowering onshore management; we further considered whether offshore engineers working in mixed onshore-offshore teams would be able to adopt agile ways of working, or whether in contrast the presumably less hierarchical Western companies would fail to ignite an agile culture in their offshore collaborations. The answers to these questions are of high importance in shaping the understanding of the compatibility of agile development methods with the use of offshoring practices.

Table 1. Prior research on culturally distinct behaviors impeding agile ways of working.

Level	Impeding behavior	References
Management behavior	Command-and-control mindset, reinforced deference to superiors	[3, 7, 13, 40-42]
	Leadership style discouraging team members from exposing problems	[37, 41]
	Leadership style discouraging team members from proposing alternatives to perceived directives from superiors	[41]
Engineers' behavior	Willingness to say yes to most requests in deference to superiors, reluctance to warn about non-feasible deadlines	[40-42]
	Reluctance to expose problems	[1, 4, 7, 37, 40, 41, 45]
	Lack of commitment to self-learning, reliance on top-down improvements	[40, 42]
	Lack of initiative to go beyond the top-down task-related instructions	[37, 45]
	Reluctance to engage in constructive disagreements and challenging discussions or voicing criticism	[1, 7, 13, 37, 42]
	Reluctance in taking responsibility for tasks and code	[37, 45]
	Reluctance to propose alternatives to perceived directives from superiors	[7, 41]

3 RESEARCH METHODOLOGY

We conducted an exploratory case study to answer our research questions [46]. We executed our study in a real-world setting and studied the collaboration between two companies (see Section 3.1). The present work is a holistic multiple-case study, in which the *context* is offshore collaboration on software projects between a mature agile company from Sweden and a consultancy company from India; each *case* is a distributed agile team (five teams were examined in our study), while the *unit of analysis* is the individual behavior of the team members [46]. In this study, the focal point is the team, as social integration happens primarily in teams and because team culture may differ from organizational culture. Further, in agile organizations, several conflicting sub-cultures [15] are generally present, for example, if a culture at an agile team level is seen as a threat because it conflicts with existing and established management habits. Therefore, there is a need to study several teams to understand the challenges of cultural incompatibilities. Finally, our data collection and analysis were divided into two steps (see Section 3.2).

3.1 Empirical Background

The context of our study is a collaboration between an outsourcer/customer company from Sweden working in the telecommunications industry and an outsourcing vendor/consultancy company from India. For confidentiality reasons, we were prohibited from disclosing the names of either of the companies.

The customer company had been using agile software development methods since 2015, and was going through another agile transformation to establish a DevOps organization. DevOps is a concept for developing software that extends agile principles to the entire software delivery process [47] and prescribes structural and procedural changes. This concept emerged from an increasing disconnect between the development and operations functions arising within large software companies [48], and is said to be a prerequisite for continuous software development by enabling knowledge sharing by breaking down barriers between development and operations, relying on automation of build, deployment, and testing systems, and focusing on shared responsibility [47]. In fact, DevOps centers rapid, flexible development iterations in which chunks of code are produced and deployed independently and supported by a high degree of automation [49], [48].

Agile and DevOps methods of working were not implemented in the studied Swedish organization alone, but also on the supplier's side, thus leading to a considerably closer collaboration between the Swedish and Indian engineers. The offshore personnel were expected to follow the same agile principles and philosophy as the

contracting organization, because cultural incompatibility across locations was seen as a potential problem, as differences in organizational culture were understood to impact the use and success of agile development practices. The latest agile transformation into a DevOps organization split across multiple global locations and companies was not without challenges. For example, it was not fully clear what level of autonomy teams in DevOps organizations should have. As a manager explained *“It is not clear where the responsibility of the team starts and ends. We need a shared understanding of how big is the DevOps [team’s] responsibility.”* Another manager explained that DevOps was part of the continuous agile transformation and that the transformation was working; however, not all external stakeholders were said to understand the transformation and were thus somewhat stuck in the old ways of thinking. As a third manager explained, *“We try to empower the teams. However, stakeholders and people outside the team are more interested in the status of the team to still feel they have control.”* To address these issues and improve collaboration and communication, Swedish employees received training in cultural differences rooted in national cultural traits. However, the training had not focused on interrelations between culture and software development processes. As a manager explained, *“Cultural courses are ok, but what do we do with [this knowledge]? So, you need to be able to reflect on the [cultural] differences.”* For this reason, the companies invited researchers to help address issues related to the interplay of national and organizational cultures across different locations.

Our investigation targeted five distributed DevOps teams composed of team members from both the Swedish company and the Indian vendor. The teams were selected with the help of the companies and represented all important business areas. The studied DevOps teams followed agile principles and ways of working relying on Scrum or Kanban (decided by each team individually), daily stand-ups, and team retrospectives as primary rituals. DevOps teams consisted of a few smaller mixed Dev and Ops teams working accordingly with tasks from the development or the operation streams, respectively, and were supported by external roles and functions typical of a DevOps organization. Each smaller team was cross-functional and involved developers, testers, a system lead or an operations lead, a team lead, and a product owner (see the profiles of the studied DevOps in Table 2). Some DevOps teams were already established, while others were more recently formed. Most of the Swedish team members had worked in the customer company for many years, while the team members from India represented a mixture of experienced members and more junior members who had recently joined the vendor company. Notably, turnover at the offshore vendor side was mentioned as an issue during our study, which we believe could be an important contextual factor.

Table 2. Profile of the research participants in the group interviews, feedback sessions, and the studied DevOps teams.

	No of Sites	Total No of Members	No of Participants	Participants		Offshore Member Roles
				Onshore	Offshore	
Preparation of the cultural workshop						
Intro	–	–	8	8	–	–
Group 1	–	–	8	8	–	–
Group 2	–	–	2	–	2	Managers (2)
Group 3	–	–	12	12	–	–
Group 4	–	–	4	–	4	Manager, Team lead, Developers (2)
TOTAL			34	28	6	
Cultural workshop sessions						
DevOps 1	2	18	18	11	7	Developers (3), Testers (2), Architect, Operations lead
DevOps 2	2	22	20	14	6	Developers (5), Team lead
DevOps 3	3	28	22	12	10	Developers (6), Testers (2), Team leads (2)
DevOps 4	2	44	20	13	7	Developers (6), Operations lead
DevOps 5	2	21	16	10	6	Consultant, Developers (4), Test lead
TOTAL		133	96	60	36	

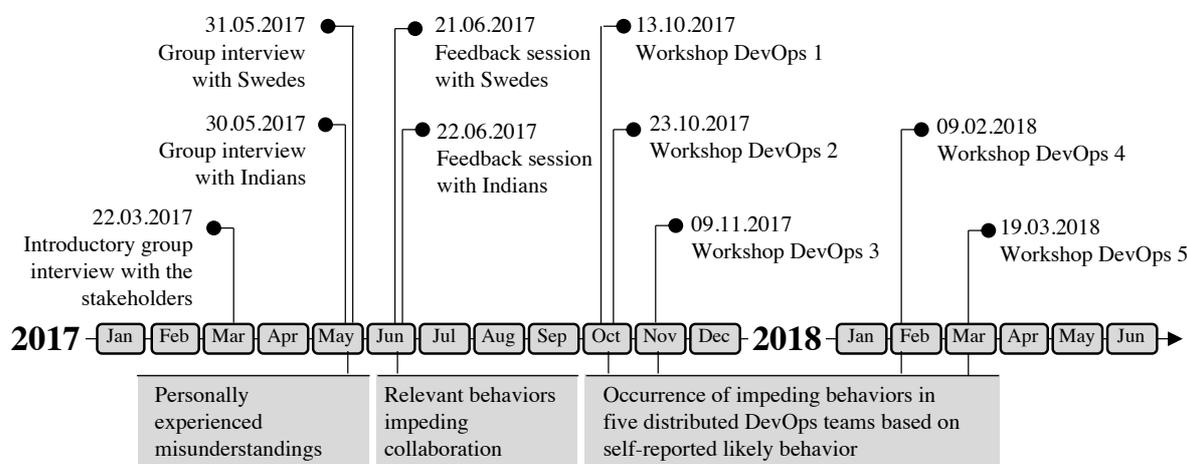


Fig. 2. Data collection activities and resulting data on the timeline

3.2 Data Collection and Analysis

A mixed approach was employed to study the impact of cultural differences on the collaboration between Swedish customers and Indian outsourcing vendors. Data collection was performed in several steps, including a quantitative data collection approach exploring recorded cultural misunderstandings and relevant behaviors impeding collaboration, and a qualitative approach checking the occurrence of impeding behaviors in the five distributed DevOps teams. The data collection activities are shown in Fig. 2 and described in the following subsections.

Introductory group interviews with stakeholders. We started the data collection with a group interview with eight stakeholders from both organizations to elicit background information about the ways of working, organizational and team structure, typical team member roles and tasks, the collaboration between the Swedish and Indian sites, and the ongoing transition to DevOps ways of working (N=8). The interviews were unstructured and were directed by two researchers. Both researchers took detailed notes during the interview, which were used to create the narrative around the empirical background of the case, verified by the representatives of both companies.

Group interviews to elicit misunderstandings. We conducted separate group interviews with representatives from Sweden and India to elicit the main sources of misunderstandings that impeded collaborative ways of working. Eight representatives participated in the session with Swedish representatives (N=8). Two experienced managers participated in an electronically mediated session with Indian representatives (N=2). The group interviews were conducted in May 2017 in English, and moderated by one of the researchers, while another researcher took detailed notes. Both sessions lasted approximately two hours and followed the same schedule – after getting to know one another and presenting the objectives, the participants were given time to connect to a web-based survey service called Mentimeter via mobile phones or computers and report personal experiences related to cultural misunderstandings. The survey form contained just one open question, and the participants were encouraged to submit as many items as possible. The submitted items were then brought up individually and discussed with all the participants in the group session. Situations in which certain misunderstandings occurred were sought, and every participant was invited to add their own reflections and bring up new ideas, which were recorded as session notes.

The generated items and recorded notes were analyzed in iterations. First, we aggregated the elicited items into one list of misunderstandings grouped by similarity. This list contained seven larger categories related to cultural differences and their impacts. Then, in the first step of analysis, we revisited each category individually, read through the session notes, and formulated items in a particular form, as follows. *As < a role and/or site*

representative > it is confusing for me when <role and/or site representative> <behavior> (when/in <situation>). In the next step of analysis, the identified 26 different confusing behaviors served as a basis for identifying situations reported as sources of misunderstandings. As a result, we drafted situations with expected and unwanted behavioral options in the format of a survey (See examples in Table 3).

Typical impeding behaviors prioritized during feedback sessions. To verify the relevance of the identified impeding behaviors and situations, we discussed our results in feedback sessions with larger groups of Swedish and Indian representatives (separately), conducted in June 2017. We invited groups of Swedish participants (N=12) and Indian participants (N=4). We also elicited the responses of the Swedish participants regarding the occurrence of impeding behaviors on the following scale, *Happens, Used to Happen, Never Happens*. This was not done in the session with Indian side representatives, because the respondents were too few and included a manager who could have influenced the results. During the feedback sessions, we learned that the given explanations of the confusing behaviors were seen as very useful; however, the participants expected to study concrete examples of solutions to overcome these challenges. We also received feedback on the relevance of behavioral options, considering expected and confusing behaviors. Further, discussing the confusing behavior among team members was also confirmed to be very useful, as it led to valuable discussion based on the developers own experience. Our survey was refined based upon the elicited feedback.

Table 3. Examples of the collected data and data analysis results.

	Cultural misunderstandings		Relevant behaviors impeding collaboration	
	Input collected in the group interviews	Analysis iteration 1	Analysis iteration 2	Refinement based on the feedback session
Swedish members	“Say ‘Yes’ without implying agreement to a subject, or silence. Lead to feeling of lack of engagement/unmet expectations.”	As an onshore product owner it is confusing for me when the offshore developers say “Yes” without implying an agreement.	#1: Communication Yes/No Onshore team lead or product owner: Can you do this? The team member:	#1: Willingness to say ‘Yes’ to most requests in deference to superiors, reluctance to warn about non-feasible deadlines.
	“At times, commitment was loosely made, like ‘EOD’, or ‘tomorrow’, with little understanding of who is waiting on other side, delay impact.”		A. Will tell PO that there is no extra time now. B. Will say ‘Maybe’ and request an approval from the local manager. C. Will say ‘Yes’, and see if they can manage other things to squeeze in this task.	A team lead or a product owner asks whether you can manage to do a task. Your spring backlog is full and there is no time for unplanned work. What is your likely response?
	“Me: can I have that (email/ report/ task done) during the day? Answer: Sure (or yes). Problem: ‘Yes’ doesn’t always mean yes.”		D. Will say ‘Yes’ and commit to the task, even if this requires working overtime.	A. Say “No, there is no extra time now.” B. Say “Only if I can skip something.” C. Say “I will do my best to accomplish it.”
Indian members	“Overpromising to get out of difficult situation.” “Indian resources not used to saying ‘No’ when asked for some tasks, especially client, led to underestimating labor required for task completion, scope creep, extended hours.”	As an offshore developer it is confusing for us to say “No” to a Product owner or a Team lead.		

In this paper, we report in detail the results relating to a selection of behaviors that were classified as hindrances to agile ways of working (see Section 4.1 and corresponding survey questions in the Appendix). These are based on the behaviors reported by both Indian and Swedish participants with occurrence scores from the Swedish session. The behaviors that were not linked to the agile ways of working are briefly mentioned at the end of Section 4.1.

Occurrence of impeding behaviors and potential social integration activities in surveyed DevOps teams

We ran five workshops with DevOps teams (mixing onshore and offshore participants; N=96) to discuss cultural differences and test the occurrence of impeding behaviors in each team (see the profiles in Table 2). The workshops

were held in the fall of 2017 and the spring of 2018. During the session, the Swedish team members participated on-site, while the Indian team members were connected via a teleconferencing system (see an illustration of the setup in Fig. 3). After explaining the purpose of the workshop and performing introductions, we asked the participants about their likely behavior in several specific situations in terms of how they would expect to act, using an electronic survey facilitated by Mentimeter. We solicited only the behavior of the participants and not their personal characteristics or experience on the team, to avoid the possibility of identifying the respondents (to increase the likelihood of obtaining honest responses). Therefore, we could not trace the recorded answers to specific individuals. After the survey, each situation, reported behaviors, and reasons for the typical behavior based on cultural studies [9] were discussed. During the discussion, team members could comment on the results of the survey, mentioning the reasons for the differences and the preferred behavior for the entire group. At the end of each workshop, identified areas of improvement, tips to overcome the identified cultural barriers, and social integration activities for the participating team were identified and discussed. One of the researchers moderated the discussion, while another took detailed notes on the conversations. In this study, we report and discuss the responses of off-shore participants (N=36) related to the recorded behaviors impeding or enabling agile ways of working. The results of the questionnaire are shown in Table 5. The narratives around each of the impeding behaviors (presented in Section 4.2) and the social integration activities (presented in Section 5.2) were constructed based on the notes taken during the workshops. Finally, the narratives, together with a draft of the manuscript of this study were sent to the case representatives for verification and approval.

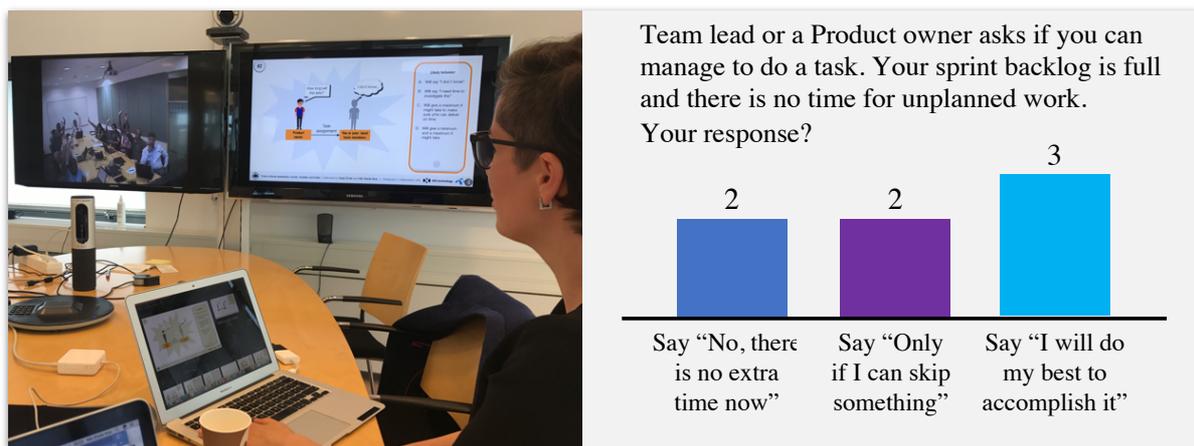


Fig. 3. Feedback session with the Indian team

4 Results

In this section, we first list the culturally distinct behaviors that were reported by Swedish and Indian team members to cause misunderstandings and impede agile ways of working in the distributed teams. We then report on the occurrence of the impeding behavior by surveying 36 Indian members from a more hierarchical organization integrated into five DevOps teams.

4.1 Behavior Impeding Agile Ways of Working

Based on the interviews, we identified 19 sources of misunderstandings reported by the Swedish participants and 14 sources of misunderstandings reported by Indian participants. Notably, both sides reported aspects they found confusing in the behavior of their counterparts as well as aspects of their own behavior that had led to

misunderstandings or confusion (as also evidenced in Table 3). Our further analysis of the situations in which differences in behavior were seen as barriers for collaboration led to an aggregated list of 26 behaviors, which was triangulated with related literature (see Table 1). Of these, 12 behaviors were prioritized as frequently occurring and important to discuss, as determined during the feedback sessions. In this study, we provide an analysis of six of these behaviors that can be classified as impediments to agile ways of working (see Table 4).

The six reported behaviors impeding agile ways of working surfaced in daily meetings, task allocation and content discussions, and team retrospectives, and all but one appeared as common sources of misunderstandings between the offshore and onshore members (see the column “Happens” in Table 4). Evidently, the most frequent impediment is the willingness to say ‘Yes’ to most requests in deference to superiors and a reluctance to warn management about non-feasible deadlines. Other impeding behaviors had varying frequency of occurrence, and “reluctance to discuss failure” almost never occurred. Some survey respondents indicated that although the impeding behaviors occurred in the past (“Used to happen” in Table 4), social integration of the Indian members led to the assimilation of the established ways of working and put an end to behavioral differences. Therefore, in the next step, we sought to further understand how common the impeding behaviors were on different teams, and what factors stimulated or supported cultural integration.

Table 3. Behavior impeding agile ways of working and the frequency of occurrence reported by the Swedish representatives (N=12).

	Happens	Used to happen	Never happened	Don't know
#1 Willingness to say yes to most requests in deference to superiors, reluctancy to warn about non-feasible deadlines	 100%			
#2 Seeking immediate manager's approval for team tasks in deference to local superiors	 75%	 17%		 25%
#3 Reluctance to reveal a lack of understanding and ask questions	 58%	 33%	 8%	
#4 Reluctance to expose problems at earliest convenience	 92%	 8%		
#5 Reluctance to discuss failure		 8%	 92%	
#6 Reluctance to voice criticism or propose alternatives to perceived directives from superiors	 67%	 8%		 25%

The other six behaviors not included in this paper were as follows.

#7 Differences in punctuality (arriving late/on-time to meetings)

#8: Switching to the Swedish language that the Indian developers did not understand for spin-off discussions

#9: Continuous reliance of Indian developers on Swedish experts

#10: Unwillingness of Swedish experts to coach Indian developers to solve the problems themselves

#11: Taking days off/leaves of absence without or with a very short prior notice,

#12: Differences in readiness to extend work hours in urgent situations

Although all these additional situations were important, we could not link them to the agile culture and practices, and thus decided not to discuss them in greater detail in this paper.

4.2 Behavior in Five Distributed Teams and Solutions Discussed

Table 5 summarizes our results from surveying offshore members from five distributed DevOps teams regarding their likely behavior in six situations (impeding behaviors are emphasized in red). Our results suggest that behaviors involving accepting unfeasible tasks in deference to superiors (#1), seeking immediate approval for tasks from management (#2), and reluctance to admit to progress lagging behind schedule (#4) were experienced by all five distributed teams, while confusing behaviors in situations #3, #5, and #6 did not appear in a few of the DevOps teams, and manifested differently in the other teams. It is also evident from the results that DevOps 1 and DevOps 5 appear to have been quite successful with the cultural integration of their Indian members, while DevOps 2, 3, and 4 were more severely challenged. This is confirmed by a more detailed analysis, which reveals that the most common respondents of impeding behavior were recently onboarded members. In the following, we further detail our findings by analyzing survey responses and notes capturing the discussions held by the team members during the sessions.

4.2.1 Willingness to say ‘Yes’ to most requests in deference to superiors, reluctance to warn about non-feasible deadlines.

Cultural barriers: This impeding behavior was the most common behavior among our respondents, with eight members in DevOps 3 reporting accepting unfeasible tasks from superiors. As a product owner (PO) from DevOps 3 revealed, *“We see [this impeding behavior] a lot, it does not just put us in a risk situation, but also other POs and stakeholders.”* Indian members from DevOps 3 explained that it was difficult for them to say “No” and therefore they were likely to use hinting words such as *“I will do my best”* in the hope that it would be interpreted correctly. An Indian participant in the feedback session explained, *“People fear that [their negative responses] will be taken personally.”* Yet, Swedes reported that they only focused on what was said, and not on what was not said.

Solutions discussed: Team leads participating in our workshops suggested that coaching Indian members to be more direct and open was important. They explained that for Swedish team members, a response of “Yes” means “I understand,” “I agree,” “I accept,” or “I approve.” Additionally, the Indian representatives from the group interview commented that there could have been additional reasons for the Indian engineers saying “Yes” to tasks, including overestimation of own capabilities and a lack of immediate understanding of the real complexity of a task and the associated expectations. This suggests that team leaders and product owners need to probe engineers’ understanding of tasks before asking for their commitment.

4.2.2 Seeking immediate manager’s approval for team tasks in deference to local superiors

Cultural barriers: This challenge was seen as a fact of life, because the Indian team members were associated with a consultancy company and conformed to local rules and regulations. In fact, the local superiors’ role in planning the offshore developers’ workloads was also evident from the fact that there were additional activities scheduled within the offshore offices, as reported by one of the participants – *“[Indian team members] might be helping some other team, or participate in some workshop. They are doing some other thing that they haven’t informed us about.”*

Solutions discussed: To overcome this challenge, many team leaders and product owners established direct communication channels with Indian managers. Therefore, this behavior was not seen as a major impediment. However, the leader of DevOps 2 was recently employed and appeared to be unaware of this difference in behavior, perhaps because of a lack of onboarding into the cultural norms of the team. Because she did not know of the

importance of maintaining regular communication with the Indian managers, she had low awareness of what was going on, which caused problems.

4.2.3 Reluctance to reveal a lack of understanding and ask questions

Cultural barriers: A few members of DevOps 2 and 3 were found to conceal a lack of understanding when discussing requirements with a product owner, and instead ask peers for help, while one respondent in DevOps 4 stated that they would proceed based on their own best assumptions. Members of DevOps 1 discussed that they experienced this challenge, especially among the new Indian members joining the team and that exchange visits and personal acquaintance between product owners and Indian team members were beneficial. Interestingly, an Indian member from DevOps 3 said that they were surprised by the results and that from his observation, people were still reluctant to ask questions. He explained one strategy to cover up for a lack of understanding: *“...when we have discussion between sites, I can see that an offshore person doesn’t understand, but doesn’t really say that. They might search for an answer in the meantime and then come back with the answer, but probably won’t say it in the meeting.”*

Solutions discussed: A product owner in DevOps 3 suggested that one way to change the current practice and then the culture was to encourage questions; he explained, *“I am used to say[ing that] if you don’t have questions, you don’t understand.”* The absence of questions as a sign that the content of the discussion might not be well understood was repeated by many. Another useful advice put on the action lists by DevOps 2 and DevOps 4 was the “Talk back” approach, which suggests that it is beneficial to avoid only informing the contractor side of plans and decisions already made to determine how a task should be approached or a problem solved, but rather ask to summarize what was said or agreed, or to explain the next steps to be taken after the discussion. The teams also agreed to avoid asking yes/no questions, such as *“Have you understood everything?”*

4.2.4 Reluctance to expose problems at earliest convenience

Cultural barriers: Another common challenge in all five teams was the fear of revealing, during team daily meetings, the fact that someone was lagging behind in their work progress. The third question in a standard daily meeting should be whether an individual notes any potential impediments to planned work progress. As an Indian member of DevOps 1 explained: *“Everyone wants to do their best, they try everything before saying that they are late.”* This reluctance to expose problems was linked to the Asian cultural concept of “losing face,” the deference to managers in typically hierarchical organizations and a “why-management” style (A “Why?” question follows when someone reveals a problem, forcing people to engage in uncomfortable explanations and thus making them “lose face”). The meetings were said to differ between Sweden and India. Team leaders in Sweden were seen as equals, while the social distance between team members occupying different “leading” roles was much larger. An Indian member participating in a feedback session explained: *“They have in mind that the [manager] is the one who decides their salary.”*

Besides the commonly expressed fears of “losing face,” Indian participants in four out of five DevOps (all except DevOps 1) described a situation in which asking for help would result in the Swedish developers “hijacking” tasks from less experienced Indian developers and solving them independently. This delicate situation was acknowledged by the Swedish counterparts. A participant from DevOps 3 explained why tasks were hijacked: *“Coaching is more time-consuming.”* However, Swedish participants from DevOps 4 justified their actions by explaining that high turnover and subsequently a lack of experience was the main cause of the problem: *“Rotation is a problem. Tasks are too complex, and thus people leave before becoming productive. Retraining offshore developers is thus seen as a burden.”*

Solutions discussed: To address issues with revealing problems and lacks of expected progress, participants from DevOps 4 suggested building trust and respect in their relationships with their Indian teammates, as well as developing more personal relationships, which is generally considered uncommon in Swedish culture. We learned that the Swedish team leads and managers in some of the DevOps groups were accepted as more accessible over time, which allowed the Indian team members more easily to raise problems, because it became less frightening to do so. To achieve that, Swedish members repeatedly emphasized that they preferred engineers to say what they thought, and not what they thought their managers wanted them to say. Engaging in polite communication was also seen as important. As someone from Sweden said, *“When we talk to offshore [teammates], we nice it up.”* Similarly, a participant from DevOps 3 explained, *“It is important to have a common understanding that you are allowed to be wrong.”* Additionally, many emphasized the importance of breaking the vicious cycle of revealing problems, then – losing tasks, then – becoming reluctant to reveal problems. DevOps teams 2, 3, 4, and 5 all wrote “Coach more” as an important action point on their To-Do lists. It was also regarded as important that the offshore developers asked for help, and especially critical for new hires.

4.2.5 Reluctance to discuss failure

Cultural barriers: This challenge was not reported as common in the studied teams, motivated by the positive atmosphere of the retrospective meetings with the Swedish team leads. An Indian member of DevOps 1 further explained: *“It depends on who is in the meeting, sometimes we keep silent. It depends on who is asking, and who is running the meeting.”* In DevOps 3, nine out of ten Indian team members reported bringing up problems for discussion. In fact, during the workshop, the Indian members said they were interested in hearing and discussing improvements beyond the current sprint, including existing knowledge gaps.

Solutions discussed: To address the reluctance to discuss failure in cases where it was an issue, it was suggested to keep local Indian managers outside the retrospectives, as their presence influenced offshore team members to be disinclined to initiate discussions about personal challenges. Further, to improve how problems were discussed, members from DevOps 3 emphasized the importance of good communication and collaboration.

4.2.6 Reluctance to voice criticism or propose alternatives to perceived directives from superiors

Cultural barriers: Similarly, challenging established ways of working during retrospectives did not appear as a common challenge. Many Indian members were likely to either challenge the development processes or propose improvements. In DevOps 2 and 3, the members who reported being reluctant to express opinions that differed from the established ways of working were more likely to be the most junior members on the teams. Participants who brought up this issue in the feedback sessions explained that Swedes who question ways of working on a daily basis were likely do so because they knew the company *“inside out,”* with many having over 20 years of experience with the company, while the Indian developers were much more inexperienced, particularly the junior developers.

Solutions discussed: One likely characteristic of collaboration that was identified as beneficial to address this issue was that of building high levels of trust between the members of the team, which some of the participants also associated with team stability. Moreover, it was discussed that expectations for proposing alternative ways of working should take into account the experience and familiarity of the team members with the development process, the company, and the team.

4.3 Other Cultural Barriers Unrelated to Agile Ways of Working

A number of cultural barriers not directly related to agile ways of working emerged in our research. These included behaviors that were said to be confusing for both Swedish team members and their Indian counterparts.

4.3.1 Late arrival to meetings, unpunctuality

Cultural barriers: In the group interviews, we learned that being late for meetings was a common trait among Indian team members. During the workshops, we learned that in India, being several minutes late was not considered inappropriate, while a common saying in Sweden states: *“If you are right on time, you are five minutes late.”* Arriving to meetings on time in Sweden is also regarded as an expression of respect to meeting attendees and their time. Evidently, based on the workshop survey results, late arrival was not a very common issue, because Indian team members working with Swedes often learned the punctuality lesson early and tried to be more on time. Interestingly, the Swedish participants reported being late often because they had many meetings and were regularly running from one room to another, which often resulted in delays, as they had a culture of ending meetings and starting new ones at the same time.

Solutions discussed: One evident solution against late arrival was to discuss the difference in attitude regarding time early. To further address late arrivals for those having many meetings in a row, members from DevOps 2 decided to switch to 45 min long meeting slots and to be more disciplined regarding ending meetings on time.

4.3.2 Late announcement of the planned leaves of absence

Cultural barriers: The problem of unexpected leaves of absence was another general issue discussed by the Swedish representatives. As someone noted, *“Emergency leave they mention a lot.”* Yet, this created challenges for the team, as another representative explained: *“Communicating absence and leave plans late affects the team negatively, as the plans for the team crash.”* During the workshops, we learned that 1) there were many more unexpected events happening in India, and 2) late announcements in India were related to a fear of local superiors. A few engineers explained that they preferred to be excused for not appearing in the office, instead of asking for a few days of absence and risking receiving a negative response from their immediate manager. On the question of what the reason for these absences was, many mentioned weddings and extended family gatherings. These differences in cultural traditions concerning wedding invitations and sizes and how many weddings a given individual commonly attends during a given time were discussed as another expression of cultural differences.

Solutions discussed: One of the key points discussed with regard to late announcements of absence was the importance of keeping commitments. Early announcements of planned absence were therefore instrumental in revisiting plans and predicting requirements for additional development labor as backup.

Table 5. Self-reported offshore member behavior in the given situations; denotes acceptable behavior while and the red color – impeding behavior

Behavior in different situations	Team 1 (N=7)	Team 2 (N=6)	Team 3 (N=10)	Team 4 (N=7)	Team 5 (N=6)
#1: Committing to tasks Willingness to say yes to most requests in deference to superiors, reluctance to warn about non-feasible deadlines					
<input checked="" type="checkbox"/> Refuse to commit to non-feasible deadlines	14%	0%	0%	29%	17%
<input checked="" type="checkbox"/> Warn about non-feasible deadlines and exchange tasks	71%	83%	20%	29%	30%
<input checked="" type="checkbox"/> Accept tasks with non-feasible deadlines	14%	17%	80%	43%	33%
15/#2: Committing to tasks Seeking immediate manager's approval for team tasks in deference to local superiors					
<input checked="" type="checkbox"/> Start working on the task directly	71%	30%	70%	14%	31%
<input checked="" type="checkbox"/> Request an approval from the immediate manager	14%	33%	30%	86%	67%
<input checked="" type="checkbox"/> Wait until the task comes from the immediate manager	14%	17%	0%	0%	0%
#3: Interaction with product owners and onshore team members Reluctancy to reveal a lack of understanding and ask questions					
<input checked="" type="checkbox"/> Ask questions directly, when not understanding something	100%	50%	0%	86%	100%
<input checked="" type="checkbox"/> Do not reveal non-understanding and seek local peer help	0%	50%	20%	0%	0%
<input checked="" type="checkbox"/> Do not reveal non-understanding and act upon assumptions	0%	0%	80%	14%	0%
#4: Daily meetings Reluctancy to expose problems at earliest convenience					
<input checked="" type="checkbox"/> Warn about the problems at the earliest convenience	86%	17%	70%	43%	83%
<input checked="" type="checkbox"/> Reluctant to expose problems	14%	83%	30%	57%	17%
#5: Team retrospectives Reluctancy to discuss failure					
<input checked="" type="checkbox"/> Bring up problems and discuss them	100%	83%	90%	100%	67%
<input checked="" type="checkbox"/> Wait until others bring up problems and then try to defend	0%	0%	0%	0%	33%
<input checked="" type="checkbox"/> Reluctant to bring up problems	0%	17%	10%	0%	0%
#6: Team retrospectives Reluctancy to voice criticism or propose alternatives to perceived directives from superiors					
<input checked="" type="checkbox"/> Voice criticism and propose process changes	N/A	33%	40%	43%	50%
<input checked="" type="checkbox"/> Voice criticism	N/A	33%	0%	0%	17%
<input checked="" type="checkbox"/> Propose changes	N/A	17%	50%	57%	33%
<input checked="" type="checkbox"/> Reluctant to voice criticism or propose process changes	N/A	17%	10%	0%	0%
#7: General Late arrival to meetings, unpunctuality					
<input checked="" type="checkbox"/> I always come to meetings on time	100%	83%	40%	57%	100%
<input checked="" type="checkbox"/> I usually come to meetings on time, but sometimes (rarely) 5-10 min late	0%	17%	60%	43%	0%
<input checked="" type="checkbox"/> I often miss the start of the meeting	0%	0%	0%	0%	0%
#8: General Late announcement of the planned leaves of absence					
<input checked="" type="checkbox"/> Plan and notify all well in advance (>1 month)	29%	50%	60%	57%	50%
<input checked="" type="checkbox"/> Plan and notify all as early as possible (but often not earlier than 1week)	57%	50%	30%	29%	50%
<input checked="" type="checkbox"/> Plan in advance, but notify all with a short notice (<1 week)	14%	0%	10%	14%	0%

4.3.3 Continuous reliance on the Swedish experts

Cultural barriers: Indian developers were said to rely on the help of Swedish experts longer than expected and repeatedly return with similar questions. However, one possible explanation for not being able to learn efficiently was that the Swedish team members were reluctant to invest their time in coaching the Indian developers (as also mentioned in Section 4.3.4). As a member of DevOps 1 explained, “*Coaching is more time-consuming*” when comparing it to solving the problems themselves. Similarly, a member of DevOps 2 apologized: “*The timeline is very stressful*” a reason that was also echoed in other sessions.

Solutions discussed: In many workshops we held, team members agreed that continuous reliance on Swedish experts was indeed related to the lack of coaching. As a result, the Swedish representatives admitted: “*We need to be more aware of [investing in] coaching,*” and that “*It is important from the company side to have scheduled time for coaching*”.

4.3.4 Confusing behaviors of the Swedish teammates

Behaviors said to be confusing for the Indian team members included the following.

- Unwillingness to coach the Indian developers: Related to the issue reported in 4.3.3, Swedish team members were perceived as hijacking tasks from the offshore team members when they were asking for help. Swedish members involved in several data collection sessions further explained that turnover in the offshore vendor company was high and the value invested in coaching the developers that would soon leave was perceived as unreasonable. At the same time, this might also explain why Indian members from some teams were reluctant to reveal a lack of understanding and ask questions, as reported in 4.2.3.
- Differences in work/life balance: Swedish team members did not demonstrate the same readiness to extend their work hours in urgent situations as the Indian team members. However, contrary to this reported information, the workshops’ results showed that the Swedish colleagues, when not forbidden by their employer, were ready to extend their work hours when needed. However, some of the participants explained that their contract did not allow this. Additionally, large differences existed in terms of typical family arrangements – Swedish team members reported having difficulties in staying for longer hours at work because of their need to pick up children from school or daycare in the absence of assistance from grandparents, which was reported as being common in India.
- Use of local language: Swedish team members were reported as switching to Swedish language to hold local discussions during joint meetings. During the workshops, we learned that this was quite common, and all participants agreed that this behavior should be avoided.
- Differences in preparing for visitors: Swedish team members were reported to have asked the Indian team members on their arrival: “*When will you be leaving?*” or “*How long will you be staying in Sweden?*” which are commonly regarded as rude questions in India, suggesting that visitors are not welcomed. During the feedback session, we learned that Swedish team members were often unaware of the visitors’ perceptions of rudeness and thus aimed to learn more about the duration of their stay to replan their agenda to be able to accommodate their guests.

5 Discussion

We have described how an agile outsourcer/customer company from Sweden working in the telecommunication industry and an outsourcing vendor consultancy company from India collaborated in a DevOps organization following agile ways of working. The focus of our study has been cultural differences explored in a number of situations, in which the habitual, expected or “normal” behavior of Indians and Swedish engineers differs. In

particular, we have reported culturally distinct behaviors that were experienced to cause misunderstandings and believed to impede agile ways of working in the distributed DevOps teams. Notably, both Swedish and Indian team members reported elements that were confusing in the behavior of their counterparts, as well as aspects of their own behavior that had led to misunderstandings or confusion. Further, we reported on the occurrence of impeding behavior by surveying 36 offshore members from the more hierarchical organization integrated into five DevOps teams, as well as eliciting the occurrence of cultural barriers from 12 representatives from Sweden. We now describe our case in light of our two research questions: “What are the barriers to agility in distributed teams with members from a hierarchical culture?” “What can agile teams do to overcome these barriers?”

Table 6. Occurrence of the cultural barriers impeding agile ways of working on the engineering level.

Impeding behavior of engineers	Prior Studies		Our Findings	
	Appeared in the related work	Appeared in the group interviews (1-2) (N=10)	Reported by the Swedish teammates in group interviews (3-4) (N=12)	Reported by the Indian teammates during the workshop (N=36)
Seeking immediate manager’s approval for team tasks in deference to local superiors		☑	Happens: 75% Used to happen: 17%	Typical for 50%
Willingness to say yes to most requests in deference to superiors, reluctance to warn about non-feasible deadlines	☑	☑	Happens – 100%	Typical for 42%
Reluctance to expose problems	☑	☑		Typical for 39%
Reluctance to reveal a lack of understanding and ask questions		☑	Happens: 58% Used to happen: 38% Never happened: 8%	Typical for 33%
Reluctance to discuss failure		☑	Used to happen: 8% Never happened: 92%	Typical for 11%
Reluctance to voice criticism or propose alternatives to perceived directives from superiors	☑	☑	Happens: 67% Used to happen: 8%	Typical for 5%
Lack of commitment to self-learning, reliance on top-down improvements	☑			
Lack of initiative to go beyond the top-down task-related instructions	☑			
Reluctance in taking responsibility for tasks and the code	☑			

5.1 Cultural Barriers Impeding the Agile Ways of Working

In response to the first research question, we first conducted a literature review, the results of which are presented in Table 1. We found three impeding management behaviors and seven impeding engineer behaviors. Through our case study, we investigated the occurrence of these cultural barriers (see the summary of our findings in Table 6 with cultural barriers sorted by their occurrence among the surveyed Indian engineers). We identified 26 behaviors rooted in national cultural differences and having a potential to lead to misunderstandings. Thus, our findings confirm previous research highlighting cultural barriers common to offshore organizations. Of these 26 behaviors, 12 behaviors were prioritized as frequently occurring and important to discuss, as they were seen as barriers to collaboration. Further, we provided an analysis of six of these behaviors that could be classified as impediments to agile ways of working and attributed to hierarchical organizational forms. These six behaviors surfaced in key agile practices and processes, such as daily meetings, task allocation and content discussions, and team retrospectives. In particular, we found the reinforcement of deference to superiors (similar to [3, 7, 13, 40-42]) to be a common barrier for the studied teams, caused by a command-and-control mindset among managers. This behavior in our case led to a tendency of engineers to say “Yes” to even unrealistic requests from their superiors (similarly to [40-42]), reluctance to expose problems (similarly to [1, 4, 7, 37, 40, 41, 45]) and a reluctance to reveal a lack

of understanding and ask questions of a superior. Even though the reluctance to reveal a lack of understanding and to ask questions was not found in all teams, we added this impeding behavior to Table 1, as it was not discovered in our literature review.

When contrasting our findings with the factors rooted in organizational culture correlating with effective use of an agile method [7, 17], we can say that the culture of the outsourcing vendor did not match the highlighted values and management style. The implication of this is that agile companies might be more likely to succeed with offshoring by establishing their own sites, working with vendors that are already familiar with agile values and ways of working, or recruiting people matching the culture of the existing teams [34]. In particular, companies may find it beneficial to assess potential vendors' management styles, in terms of, facilitative leadership, collaboration-oriented management style, and focus on empowerment as crucial factors for adopting agility [17].

5.2 Overcoming the Cultural Barriers When Working with Offshore Members from a Hierarchical Organization

The first step in understanding what solutions we can find to overcome cultural barriers is to understand whether it is possible at all. Some researchers have suggested that cultural barriers are likely to remain because the major differences in norms and values cannot be harmonized, as they derive from deep-seated differences in cultural background, education, and working life [13]. However, our findings suggest that behavior seems to change when engineers are exposed to a culturally distinct environment. When continuously encouraged and exercised, the new ways of acting become habitual, expected and “normal”, and “go without saying”. The changes over time might be evidenced by the fact that not all impeding behaviors were found in all teams or among all members. For example, the reluctance to reveal a lack of understanding and ask questions of a superior was found in three out of five teams, while reluctance to discuss failure almost never happened, and some impeding behaviors only happened in the past. One explanation could be that, even though a hierarchal culture was present, some behaviors had changed over time. Such an explanation is consonant with the study by Stray et al. [22] which found that agile practices changed over time in a globally distributed project and that by Ivari [16], who argued that the relationship between an organizational culture and agile ways of working is dynamic and therefore should be expected to continuously evolve.

Potential solutions can be classified into two major categories, including altering collective behaviors when encountering cultural differences, and socially integrating the members of the different cultures. With respect to the Swedish members' behavioral changes, we found, for example, that product owners should avoid asking yes/no questions, should use a “talk back” approach to check whether offshore engineers understood their assigned tasks, team leads should establish direct communication with the immediate managers offshore and potentially alter the task assignment processes. On the other hand, our findings also demonstrate that Swedish members in several teams succeeded in integrating members of a significantly different culture who were working within a hierarchical business organization, and stimulating changes in their behavior. This was done, for example, by encouraging Indian engineers to be direct, demonstrating that discussing failure was important and failures did not necessarily lead to punishment.

However, our experience shows that distributed teams were often left to experiment and adjust their ways of working, because cultural awareness is often gained only with experience. This has been noted by Casey [14], who found that the importance of and requirement for cultural training was often not recognized before significant time, effort, and resources were wasted. Therefore, we recommend companies and distributed teams engage in cross-cultural communication courses to discuss values that should govern behavior, as suggested in [50]. The importance of such training is confirmed in one of our findings that the most common respondents stating that they engaged in impeding behaviors were recently onboarded members. Although training is understood to be important, recruiting personnel who match the culture of existing teams is also known as a factor in project success [34]. Perhaps the most important task is to make the Indian developers stay in a given job position and reduce

turnover, as discussed by many Swedish team members, who said they were reluctant to train their Indian counterparts and preferred to perform the challenging tasks themselves.

This perception of tasks being hijacked and the temptation to solve a problem instead of coaching the less-experienced offshore team members suggests a need for more external leaders and coaches, and better coordination among them. Even though the need for external leaders is reduced in agile teams, there are still reasons why external leadership is important in agile software development [29]. Our findings demonstrate the need for external leaders and coaches. Coaching Indian members to be direct and open was essential, and a lack of time to coach others was also an explanation for why tasks were hijacked. Furthermore, when an engineer started losing tasks because of this phenomenon, consequently people stopped reporting problems. Therefore, further direct communication between managers seemed to be a key to revealing such problems early.

5.3 Implication for Practice

Agile software development emphasizes that teams should be self-managed or self-organized. However, agile methods offer no advice on how to implement them. In a similar context, Dorairaj and Nobel [51] found that collective focus on working together as a team was a key enabler and that building trust over time bridged cultural differences. Furthermore, coordinating work by constant feedback and increasing trust in distributed projects was found to be beneficial [52]. Our study confirmed these findings. For team leads and onshore managers struggling to conduct an agile transformation and to overcome the same cultural barriers as those reported in this article, we recommend the following.

- Establish good communication channels with local offshore managers and agree on an efficient task allocation procedure.
- Schedule more frequent check-ins or updates.
- Create an environment of psychological safety and trust.
- Set an example by taking ownership for failures safely, show how teams can learn from mistakes, and reward people for making mistakes if/when they lead to valuable lessons.
- Seek out *one-to-one* conversations.
- Encourage offshore members to be more direct.
- Use “Talk Back” approach to check the understanding.
- Encourage suggestions of better ways of working.
- Do not criticize ideas; compliment people for valuable input.
- Set an example of changes that led to action.
- Avoid offshore managers’ presence in team retrospectives and meetings where honest and open input and feedback from offshore members is important.
- Let external leaders focus on the coaching function and removing impediments.
- Pay special attention to the socialization and cultural training in the onboarding of new team members.

5.4 Limitations and Threats to Validity

In this subsection, we discuss the limitations and issues that might threaten the validity of our results. We conducted a study in the context of offshore collaboration between a mature agile company from Sweden and a consultancy company from India. Given the qualitative nature of our study, in the following we discuss validity, reliability, and generalizability threats following the guidelines of Leung [53].

Validity refers to the appropriateness of the method. While we report on the cultural barriers to being agile, our findings represent the experiences from one case of an offshore collaboration between a mature agile company from Sweden and a consultancy company from India. Further, our data collection relied on open-ended inquiries. Thus, the identified cultural barriers may not be exhaustive, nor cover all known barriers. In Table 6, we show a

comparison of our findings with the related literature, illustrating the overlap. Next, our findings are biased towards the Swedish perspective in this study, as the number of Swedish participants in our research was much higher than that of the Indian participants. The premise of our study is that the hierarchical business management models commonly occurring within Indian culture impede the adoption of agile ways of working. We do not make the same assumption about Swedish business culture or explore the organizational culture of the case company in Sweden, which is a limitation of our study. Several steps were made to ensure the validity of our findings regarding the behavior of Indian counterparts working in mixed teams. We designed the empirical study in a two-stage fashion, aiming to improve objectivity when formulating and selecting situations and impeding behaviors. Our results are both empirically based (misunderstandings gathered in group interviews and social integration activities elicited during the workshops) and empirically validated (impeding behaviors prioritized during feedback sessions and their occurrence surveyed during the workshops). We also used an anonymized data collection tool and obfuscated the results to ensure the anonymity of the respondents, to eliminate unwillingness to report personal confusion.

Reliability in qualitative research refers to the replicability of the results. A margin of variability in results is accepted when dealing with qualitative research [53] or mixed methods, because the subjectivity of the researchers is embedded in the roots of the analysis. The main threat is thus related to consistency. To mitigate this threat, we let the participants report their responses in a data collection tool and took care to be systematic when taking notes and documenting the discussions during the sessions, and retain quotations from the participants as precisely as possible. Furthermore, we conducted feedback sessions to validate our interpretation of impeding behaviors and situations.

Possible threats to the **validity** and **reliability** of our results are related to the reluctance of Indian participants to express their opinions in front of managers or Swedish peers, as well as reluctance to talk about compromising issues. To alleviate these threats, we separated the Indian and Swedish participants in the initial group interviews and feedback sessions to be able to speak more openly. In the mixed workshop sessions, we explained the importance of truthful responses and asked our Swedish contacts from the consultancy company to encourage openness prior to the workshops. We also used a survey form that allowed participants to provide their responses anonymously, which remained untraceable to individuals even during the discussions. The reported problematic behavior in the survey, confirming comments received from the Swedish participants and reflections voiced by more experienced Indian participants (i.e., triangulation of the data sources) lead us to believe that we have elicited responses that were as honest and open as possible.

Data triangulation is the core principle of case study research [46]. To enhance the **validity** and **reliability** of our results, the individual responses elicited during the group interviews were first discussed in the respective groups and compared across the onshore and offshore groups. We then elicited quantitative data from larger groups of participants (survey responses) to minimize bias toward selected individuals. The quantitative data elicited through the survey were further triangulated with qualitative data (notes) from the discussions held during the workshops.

Generalizability of the conclusions drawn from our results are, of course, limited to the studied context. The empirical results focusing on the behavioral differences and situations could be applicable and interesting for other Scandinavian companies working with India (because Scandinavian countries belong to a relatively homogeneous cultural cluster [10]), but do not apply to relationships between other pairs of countries or Scandinavians working with other offshoring destinations than India. Furthermore, because our research is based on a single case, it is possible that the behavior of the Swedish and Indian members is only representative of the studied case, because it could depend on the organizational culture, and thus not be generalizable to the national culture. We have addressed this threat by comparing our results with related research and highlighting case-specific findings (see Table 6). At the same time, the general finding disproving the existing view that cultural barriers are likely to remain because the major differences in norms and values cannot be harmonized [13] is likely to have broader generalizability. We believe that it is fair to assume that the gradual changes in behavior that we observed as a result of the experience gained in working in a mixed environment may also occur in similar contexts in other organizations.

5.5 Future Work

This study reports cultural barriers in a collaboration between an agile company from Sweden and an Indian consultancy company with hierarchical ways of working. We look at culture from a behavioral perspective, i.e., habitual ways of acting in certain situations, and capture the changes in behavior over time by identifying both common cultural barriers and those that used to happen in the past. Accordingly, there are several possible directions for further work.

First, we believe that more research is needed in other companies in similar contexts (Swedish and agile companies collaborating with Indian companies with hierarchical business culture) to validate the already identified cultural barriers. Second, our study could be replicated in other contexts to understand the generalizability of our findings (agile companies in other western regions collaborating with companies with hierarchical business culture from other Asian countries). Third, more research is needed to better understand the dynamic nature of the cross-cultural collaborations and how the behavior evolves over time. Thus, we encourage future studies that are conducted over an extended period of time, evaluating the time it takes and ways to bridge the cultures, specifically addressing the question of social integration of the new hires. Finally, as culture is a complex and multifaceted concept (it can be attributed to a nation, an organization, a group, or even an individual), there is a need for understanding these different levels and the interplay between them. In our research, we have assumed that agile organizations in a particular national region and organizations with hierarchical business culture from a particular national region have a certain influence on individual behavior, but future research is needed to critically evaluate these assumptions and the association between the national, organizational, group and individual cultures.

6 Conclusions

Distributed collaboration remains challenging and managing distributed projects with agile ways of working is even more so, but these challenges are even further amplified when the distributed collaboration also clashes with cultural barriers. In this study, we explore barriers rooted in the differences between national and organizational cultures. Our results confirm the existing research that organizational and national cultural barriers may impede collaboration in general and the successful functioning of the agile ways of working in particular. However, our empirical findings from studying the behavior of offshore members from five distributed DevOps teams also suggest that their behavior changed over time as they were exposed to the agile ways of working and to flatter organizational culture, despite the fact that their local organizational culture remained radically different, that is, highly hierarchical. This was possible because the Swedish team members altered their behavior and employed some tactical approaches to overcome the cultural barriers, as well as because the Indian team members changed their behavior and socially integrated into the agile ways of working. In other words, when continuously encouraged and exercised, the new ways of acting became habitual, expected and “normal”. Further, we found that onboarding new team members into the cultural norms of the team is an important task, which might be quite frequent in offshore companies where staff turnover is high. Our study resulted in a list of recommendations for companies willing to discuss cultural differences and foster cultural integration in distributed projects.

Acknowledgement

This research was funded by the GoLD and ScaleWise projects and the Swedish Knowledge Foundation under the KK-Hög grants 2016/0191 and 2019/0087, and by the A-team project and the Research Council of Norway through grant 267704.

References

- [1] N.B. Moe, D. Smite, G.K. Hanssen, H. Barney, From offshore outsourcing to insourcing and partnerships: four failed outsourcing attempts, *Empirical Software Engineering*, (2013) 1-34. <https://doi.org/10.1007/s10664-013-9272-x>
- [2] D. Šmite, N.B. Moe, P.J. Ågerfalk, Agility Across Time and Space: Implementing Agile Methods in Global Software Projects, in: D. Šmite, N.B. Moe, P.J. Ågerfalk (Eds.), Springer-Verlag, Berlin, Heidelberg, 2010.
- [3] H. Holmström, B. Fitzgerald, P.J. Ågerfalk, Q.E. Conchuir, Agile Practices Reduce Distance in Global Software Development, *Information Systems Management*, 23 (2006) 7 - 18.
- [4] V. Stray, N.B. Moe, Understanding coordination in global software engineering: A mixed-methods study on the use of meetings and Slack, *Journal of Systems and Software*, 170 (2020) 110717. <https://doi.org/10.1016/j.jss.2020.110717>
- [5] J. Iivari, N. Iivari, The relationship between organizational culture and the deployment of agile methods, *Information and Software Technology*, 53 (2011) 509-520.
- [6] R.A. Guzzo, M.W. Dickson, Teams in organizations: Recent research on performance and effectiveness, *Annual Review of Psychology*, 47 (1996) 307-338.
- [7] R. Hoda, J. Noble, S. Marshall, Supporting self-organizing agile teams, in: *International Conference on Agile Software Development*, Springer, 2011, 73-87.
- [8] J. Van Maanen, A. Laurent, The flow of culture: some notes on globalization and the multinational corporation, in: *Organization theory and the multinational corporation*, Springer, 1993, 275-312.
- [9] G.H. Hofstede, G.J. Hofstede, M. Minkov, *Cultures and organizations: Software of the mind*, McGraw-hill New York, 2005.
- [10] E. Carmel, R. Agarwal, Tactical approaches for alleviating distance in global software development, *IEEE Software*, 18 (2001) 22-29. <https://doi.org/10.1109/52.914734>
- [11] E.T. Hall, The silent language in overseas business, *Harvard Business Review*, 38 (1960) 87-96.
- [12] M. Gallivan, M. Srite, Information technology and culture: Identifying fragmentary and holistic perspectives of culture, *Information and Organization*, 15 (2005) 295-338.
- [13] S. Krishna, S. Sahay, G. Walsham, Managing cross-cultural issues in global software outsourcing, *Communications of the Acm*, 47 (2004) 62-66.
- [14] V. Casey, Leveraging or exploiting cultural difference?, in: *Fourth IEEE International Conference on Global Software Engineering*, IEEE, 2009, 8-17.
- [15] K. Kautz, C.F. Pedersen, O. Monrad, Cultures of agility—Agile software development in practice, in: *20th Australasian Conference on Information Systems*, Melbourne, 2009, 174-184.
- [16] N. Iivari, Culturally compatible usability work: An interpretive case study on the relationship between usability work and its cultural context in software product development organizations, *Journal of Organizational and End User Computing (JOEUC)*, 22 (2010) 40-65.

- [17] D.E. Strode, S.L. Huff, A. Tretiakov, The impact of organizational culture on agile method use, in: 42nd Hawaii International Conference on System Sciences, IEEE, 2009, 1-9.
- [18] N.B. Moe, T. Dingsøy, T. Dybå, Overcoming Barriers to Self-Management in Software Teams, *IEEE Software*, 26 (2009) 20-26.
- [19] N.B. Moe, T. Dingsøy, T. Dybå, A teamwork model for understanding an agile team: A case study of a Scrum project, *Information and Software Technology*, 52 (2010) 480-491.
- [20] L. Williams, What agile teams think of agile principles, *Communications of the Acm*, 55 (2012) 71-76.
- [21] N.B. Moe, B. Dahl, V. Stray, L.S. Karlsen, S. Schjødt-Osmo, Team autonomy in large-scale agile, in: 52nd Hawaii International Conference on System Sciences, 2019, 6997-7006.
- [22] V. Stray, T.E. Fægri, N.B. Moe, Exploring norms in agile software teams, in: *International Conference on Product-Focused Software Process Improvement*, Springer, 2016, 458-467.
- [23] J. Erickson, K. Lyytinen, K. Siau, Agile Modeling, Agile Software Development, and Extreme Programming: The State of Research, *Journal of Database Management*, 16 (2005) 88 - 100.
- [24] J.R. Katzenbach, D.K. Smith, The Discipline of Teams, *Harvard Business Review*, 71 (1993) 111-120.
- [25] J.R. Hackman, The psychology of self-management in organizations, in: M.S. Pallack, R.O. Perloff (Eds.) *Psychology and work: Productivity, change, and employment*, American Psychological Association, Washington, DC, 1986.
- [26] C. Argyris, D.A. Schön, *Organizational learning II: Theory, method, and practice*, Addison-Wesley, Reading, Massachusetts, 1996.
- [27] G. Morgan, *Images of Organizations*, SAGE publications, Thousand Oaks, CA, 2006.
- [28] R. van Solingen, E. Berghout, R. Kusters, J. Trienekens, From process improvement to people improvement: enabling learning in software development, *Information and Software Technology*, 42 (2000) 965-971. [https://doi.org/10.1016/s0950-5849\(00\)00148-8](https://doi.org/10.1016/s0950-5849(00)00148-8)
- [29] F.P. Morgeson, The external leadership of self-managing teams: intervening in the context of novel and disruptive events, *Journal of Applied Psychology*, 90 (2005) 497.
- [30] G.A. Yukl, *Leadership in organizations*, s ed., Prentice Hall, Upper Saddle River, NJ, 2002.
- [31] D.G. Ancona, Outward bound: strategic for team survival in an organization, *Academy of Management Journal*, 33 (1990) 334-365.
- [32] J.R. Hackman, R. Wageman, A theory of team coaching, *Academy of Management Review*, 30 (2005) 269-287.
- [33] K. Dikert, M. Paasivaara, C. Lassenius, Challenges and success factors for large-scale agile transformations: A systematic literature review, *Journal of Systems and Software*, 119 (2016) 87-108. <http://dx.doi.org/10.1016/j.jss.2016.06.013>

- [34] N.B. Moe, V. Stray, M.R. Goplen, Studying onboarding in distributed software teams: a case study and guidelines, in: *Evaluation and Assessment in Software Engineering*, 2020, 150-159.
- [35] T. Dingsøy, S. Nerur, V. Balijepally, N.B. Moe, A decade of agile methodologies: Towards explaining agile software development, *Journal of Systems and Software*, 85 (2012) 1213-1221. <http://dx.doi.org/10.1016/j.jss.2012.02.033>
- [36] B.H. Liu, H. Zhang, L.M. Dong, Survey on State of DevOps in China, *Journal of Software*, 30 (2019) 3206-3226. <http://dx.doi.org/10.13328/j.cnki.jos.005796>
- [37] M. Kajko-Mattsson, G. Azizyan, M.K. Magarian, Classes of distributed agile development problems, in: *2010 Agile Conference*, IEEE, 2010, 51-58.
- [38] S. Dorairaj, J. Noble, G. Allan, Agile software development with distributed teams: Senior management support, in: *2013 IEEE 8th International Conference on Global Software Engineering*, IEEE, 2013, 197-205.
- [39] H. Ozawa, L. Zhang, Adapting agile methodology to overcome social differences in project members, in: *2013 Agile Conference*, IEEE, 2013, 82-87.
- [40] H. Ayed, B. Vanderose, N. Habra, Agile cultural challenges in Europe and Asia: Insights from practitioners, in: *IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track, ICSE-SEIP*, Buenos Aires, Argentina, 2017, 153-162.
- [41] M. Fowler, Using an agile software process with offshore development, <http://martinfowler.com/articles/agileOffshore.html>, (2006).
- [42] M. Summers, Insights into an Agile adventure with offshore partners, in: *Agile 2008 Conference*, IEEE, 2008, 333-338.
- [43] S. Sundararajan, M. Bhasi, P.K. Vijayaraghavan, Case study on risk management practice in large offshore-outsourced Agile software projects, *IET software*, 8 (2014) 245-257.
- [44] J. Srinivasan, K. Lundqvist, Agile in India: Challenges and lessons learned, in: *3rd India software engineering conference*, 2010, 125-130.
- [45] S. Hole, N.B. Moe, A Case Study of Coordination in Distributed Agile Software Development, in: *Software Process Improvement*, 2008, 189-200.
- [46] P. Runeson, M. Host, A. Rainer, B. Regnell, *Case study research in software engineering: Guidelines and examples*, John Wiley & Sons, 2012.
- [47] R. Jabbari, N. bin Ali, K. Petersen, B. Tanveer, Towards a benefits dependency network for DevOps based on a systematic literature review, *Journal of Software: Evolution and Process*, (2018). 10.1002/smr.1957
- [48] B. Fitzgerald, K.-J. Stol, Continuous software engineering: A roadmap and agenda, *Journal of Systems and Software*, 123 (2017) 176-189.
- [49] C. Ebert, G. Gallardo, J. Hernantes, N. Serrano, *DevOps*, IEEE Software, 33 (2016) 94-100.

[50] M.M. Lehman, Uncertainty in Computer Applications and its Control through the Engineering of Software, Software Maintenance: Research and Practice, 1 (1989) 3-27.

[51] S. Dorairaj, J. Noble, Agile software development with distributed teams: Agility, distribution and trust, in: 2013 Agile Conference, IEEE, 2013, 1-10.

[52] N.B. Moe, D. Smite, Understanding a lack of trust in Global Software Teams: a multiple-case study, Software Process: Improvement and Practice, 13 (2008) 217-231. <http://dx.doi.org/10.1002/spip.v13:3>

[53] L. Leung, Validity, reliability, and generalizability in qualitative research, Journal of family medicine and primary care, 4 (2015) 324.

Appendix: Survey questions

The following are the questions given to the survey respondents to detect their likely behavior in different situations. Note, the symbol determines acceptable behavioral options, while determines impeding behavioral options.

#1: Willingness to say yes to most requests in deference to superiors, reluctance to warn about non-feasible deadlines.

A team lead or a product owner asks whether you can manage to do a task. Your spring backlog is full and there is no time for unplanned work. What is your likely response?

- Say "No, there is no extra time now"
- Say "Only if I can skip something"
- Say "I will do my best to accomplish it"

#2: Seeking immediate manager's approval for team tasks in deference to local superiors.

A team lead or a product owner assigns a task to you. What is your likely action?

- Start working on the task directly
- Request an approval from the immediate manager
- Wait until the task comes from the immediate manager

#3: Reluctance to reveal a lack of understanding and ask questions.

A team lead or a product owner gives a brief task description and asks if the task is clear to you. You understand that a lot of details are unclear. What is your likely action?

- Say "No" and ask questions directly
- Say "Yes" and later ask peers for clarification
- Say "Yes" and make the best assumption to complete the task

#4: Reluctance to expose problems at earliest convenience.

Team lead in a daily standup asks if you experience problems, if you are on time. You are delayed, there is a high chance you will not meet deadline. What is your likely action?

- Confess that you are lagging behind the schedule
- Say "I am doing my best to meet the deadline"
- Say "No, I don't have any problems"

#5: Reluctance to discuss failure.

In the last sprint the team underperformed. One of the reasons for this was that you have not delivered what was expected or planned. A team lead in a retrospective meeting asks everyone what went wrong and why. What is your likely reaction?

- Bring up the problem and analyze it
- Wait until others point out this problem and then try to defend
- Keep silent

#6: Reluctance to voice criticism or propose alternatives to perceived directives from superiors.

Team lead asks in a retrospective whether the established process works and expects change suggestions. What is your likely behavior?

- Challenge the process and propose changes

28

- Challenge the process
- Propose changes
- Prefer to keep silent

#7: Late arrival to meetings, unpunctuality.

Your attendance of the joint meeting can be characterized best with the following answer:

- I always come to meetings on time
- I usually come to meetings on time, but sometimes (rarely) 5-10 min late
- I often miss the start of the meeting

#8: Late announcement of the planned leaves of absence.

How do you plan your days of and how do you communicate them to the team?

- Plan and notify all well in advance (>1 month)
- Plan and notify all as early as possible (but often not earlier than 1 week)
- Plan in advance, but notify all with a short notice (<1 week)