AGILE DISTRIBUTED SOFTWARE DEVELOPMENT IN NINE CENTRAL EUROPEAN TEAMS: CHALLENGES, BENEFITS AND RECOMMENDATIONS

Manuel Stadler, Raoul Vallon, Martin Pazderka and Thomas Grechenig

Research Group for Industrial Software, Vienna University of Technology,
Vienna, Austria

ABSTRACT

Although initially designed for co-located teams, agile methodologies promise mitigation to the challenges present in distributed software development with their demand for frequent communication. We examine the application of agile practices in software engineering teams with low geographical distribution in Austria and Germany. To gather insights on challenges and benefits faced by distributed teams we conduct interviews with eleven representatives and analyse the interview transcripts using the inductive category formation method. As a result, we identify four major challenges, such as technical obstructions or the impediments different language abilities have on communication, and four benefits, regarding collaboration and information radiation, that agile methods yield in distributed teams. Based on our analysis of challenges and benefits, we deduct seven recommendations to improve collaboration, overcome distance and avoid pitfalls. Key recommendations for teams with low geographical distance include that teams should get together at certain points to build relationships and trust and share information face-to-face.

KEYWORDS

Agile Distributed Software Development, Distributed Agile, Nearshoring, Agile Methods

1. INTRODUCTION

Agile methods continually gain attention and face extensive adoption but there is also a trend towards distribution, where teams facilitate modern communication technologies to bridge the challenges of spatial dispersion [1]. In 2016 around 25% of German software developers reported to have experience with working remote [2]. Kajko-Mattsson et al. [1] summarise the following: "Being in stark contrast with each other, Agile and Distributed Software Development (DSD) methods are regarded as partners in an impossible marriage. Despite this, many organisations consider them as practices worth striving for".

This paper examines how teams distributed within a low geographical distance apply agile methods, which challenges were faced, and which benefits teams experienced. We start with exploring aspects of distribution, like different dimensions of distance and challenges they introduce, as well as the general state of the art of agile literature in Section 2. Section 3 then displays the methodology of our research, enumerates the research questions and propositions, and presents the analysed teams. Section 4 presents the results from our cross-case analysis, followed by a discussion of the research questions. Section 5 summarises our findings, lists our derived recommendations and closes with future perspectives.

2. BACKGROUND

A team is considered distributed when team members are sited in different, geographically dispersed, locations. By definition, it is not necessary that every team member is remote, at least one person has to be separated from the rest of the team [3]. The reasons for a team to work in a distributed setting are manifold, a regional argument is talent acquisition [4] or accessing local knowledge [5]. On a global scale, cost advantages are driving arguments for off shoring work to remote countries and continents [6], or accessing new markets and being near to customers [7].

Implementing agile methodologies in distributed teams does not seem very applicable at first glance. Agile frameworks strongly demand co-location of teams and while not being an inevitable requirement, experts strongly recommend to co-locate team members as close as possible (e.g. [8] or [9]). On a closer look the term co-location is not specifically mentioned in the agile manifesto, but is often derived from one of its 12 principles:

"The most efficient and effective method of conveying information to and within a development team is face-to-face conversation." [10]

Agile values are as suited for distributed teams as they are for co-located ones, as Beck and Andres [8] state: "the values of XP are just as suited to multi-site development as they are to teams that sit together". Cohn [11] furthermore argues in a similar way about a specific agile methodology: "Scrum's preference for face-to-face communication, the argument goes, makes it a poor choice for distributed teams. Fortunately, this argument is false".

2.1 DIMENSIONS OF DISTANCE

Next to geographical separation, there are also cultural, temporal and configurational dimensions to consider [3].

Geographical Distance can be measured directionally and is described as "the effort required for one actor to visit another at the latter's home site" [3]. The Allen-Curve states that "the probability that people in a given organisation will communicate with each other declines precipitously the farther away from each other they are situated and reaches an asymptotic level at about 50 meters". This is not only valid for face-to-face communication, but the data from the study also shows a decline in the usage of all communication media with increasing distance [12].

Socio-cultural Distance is one of the most complex and therefore least understood dimensions of distance, but a critical element in a distributed setting [13]. It involves national as well as organisational culture, work ethic and motivation, as well as spoken languages. Two individuals with different national backgrounds may share a common organisational culture and therefore have a low socio-cultural distance. A low socio-cultural distance lowers risks and increases communication [3]. In the scope of the same company, sites still often differ in various aspects like informal habits, practices or processes [14].

Configurational Dimension is defined by O'Leary and Cummings as "the arrangement of members across sites independent of the spatial and temporal distances among them [15]. Especially situations with a concentrated core team and isolated members on different sites decrease awareness towards remote colleagues. A larger number of sites also boosts complexity for coordination and raises conflict potential [15].

Temporal Distance, in terms of different time zones, impedes synchronous communication between individuals and therefore has a negative impact on communication [16]. In our study this aspect is not relevant as it only contains teams within the same time zone.

2.2 CHALLENGES OF DISTANCE

Those dimensions of Section 2.1 in turn induce various challenges. To be successful, it is necessary to have control [17] and coordination [18], both factors driven by communication.

Coordination is the challenge of "managing interdependencies, uncertainties and equivocalities, conflicts, technology representations, and their interrelations" [19]. From a management perspective, it is the process of integrating tasks with organisational units to enable those entities to contribute value to the all-up objective. This process of integration commonly requires regular and intense communication [16].
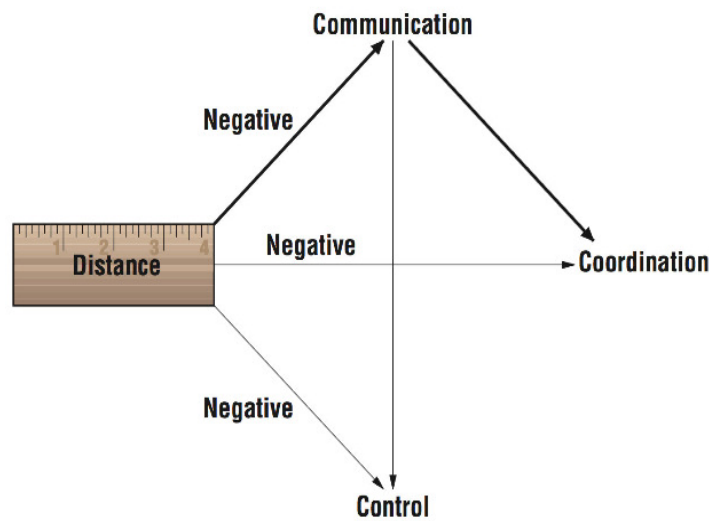
Figure 1. Impacts of distance [16]

The control aspect can be split into two categories: formal and informal control [20]. Self-organisation of agile teams taps into intrinsic motivation of team members and is a form of informal control. The commitment of individuals towards the project goal can rise due to individual motivations and allegiance towards a mutual goal [11]. Communication, the coupling factor between coordination and control, is the exchange of information between a sender and a receiver with the goal of reaching a mutual understanding. Figure 1 shows that distance directly impedes coordination and control as well as indirectly through its restraining impact on communication, while the bold arrows "represent the main challenge of global software development" [16].

3. METHODOLOGY

Building on the literature review, semi-structured interviews constitute the empirical foundation of this paper, the full approach is depicted in detail in Figure 2. Our chosen approach can be classified as evaluative qualitative research which is "particularly adept at looking at the dynamics of how things operate" [21].

3.1 OBJECTIVE AND CONSTRAINTS

We deliberately chose a low geographical distance to examine the particular challenges and benefits of that confined scenario as a lot of recent research, in contrast, focuses on globally distributed software development. To define the boundaries we use the taxonomy from Šmite et al. [22] as a starting point, focusing on teams that can be classified as either *Onshore - Insourcing - Geographically Close* or *Offshore - Insourcing - Geographically Near*. Locations therefore can be in the same country, or - if spanning multiple countries - situations where the time zone offset is not more than 4 hours and sites are reachable within a reasonable amount of time (where, if air travel is necessary, the flying time is less than 2 hours). We furthermore confined the boundaries and defined the following list of criteria which teams and experts have to meet:

- Having at least one permanent team member who is not located with the rest of the team.
- At least one office or site has to be located in Austria or Germany in order to be easily accessible to the authors. Other sites have to be located in countries adjoining Austria.
- Follow a defined agile process.
- Interviewed experts have to be directly involved in the teams and hold a senior position or a leading role.
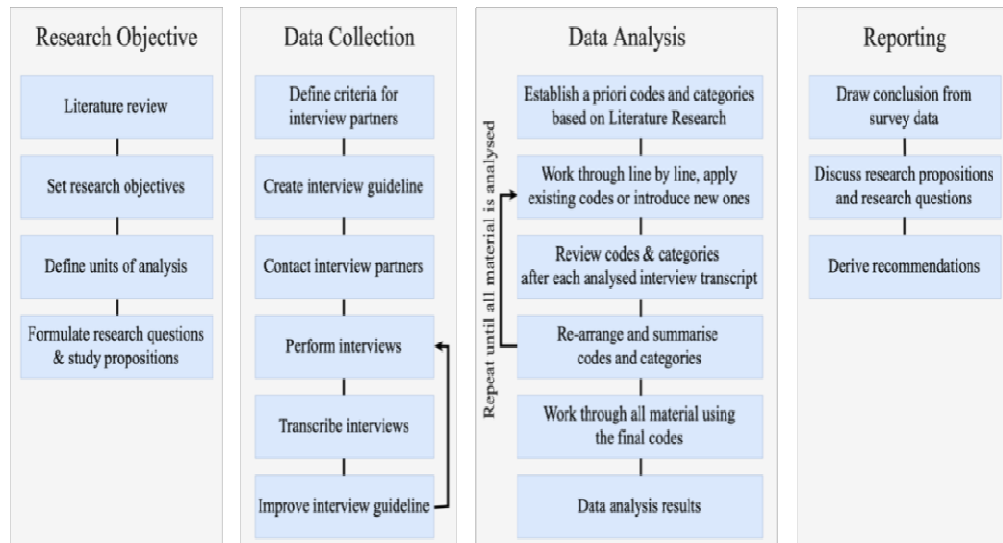

Figure 2. Research design

3.2 RESEARCH QUESTIONS AND PROPOSITIONS

Based on the reviewed literature and the defined boundaries, we formulated our primary research question as well as two sub-questions to narrow the field of investigation further:

- RQ: How are agile processes used in a distributed team (limited to a low geographical and time dispersion)?

- RQ.A: Which challenges have to be faced when using agile methods and how can those issues be handled?

- RQ.B: Which benefits result from pursuing agile methods in a distributed setting for a team?

The following propositions are intended to highlight different aspects of distributed agile teams and are derived from the research questions and the literature review.

They provide structure and detail to the enquired topic by serving as more particular "implementations" of research questions [23] as well as directing attention to certain aspects within the research topic [24]. Each proposition is followed by the rationale, why we chose to create the respective proposition.

P1: Iterations in distributed agile teams are similar in length to iterations in co-located teams: Short iteration cycles which re-evaluate the current status of a team and project are used to overcome coordination and control challenges. We chose this proposition to check the difference in iterations between co-located and distributed teams.

P2: Modern software and tools are a major factor for the success of distributed teams: This proposition assumes that project management software as well as digital communication tools designed for distributed collaboration are essential and a precondition for effective work. This importance is also argued in the fourth conjecture of Layman et al. [25].

P3: Aside from communicative skills there are no special requirements for team members in distributed teams: Because distance negatively impacts communication, team members in distributed teams need to be even more aware of its necessity. We ask if there are any other specific skills necessary to participate in a distributed team successfully.

P4: While for short, standardised meeting situations remote communication is sufficient, face-to-face communication is preferred in longer, informal situations: While modern digital communication tools are sufficient for shorter formalised situations (like a daily standup), we want to find out if longer and more informal situations (like retrospectives) are nevertheless preferably organised in co-located gatherings.

P5: Technical faults and limitations pose a serious issue for distributed communication: Infrastructure problems like a failing WiFi or a slow internet connection can massively intensify the primary challenges of remote communication.

P6: Informal and frequent communication improves collaboration between remote colleagues: This proposition aims to investigate the importance of interaction and informal communication between individuals as encouraged by the agile manifesto [26].

3.3 DATA COLLECTION

The interview guideline was designed to gather specific experience and approaches that the interviewees or the team they speak for encountered, as well as general thoughts on distributed agile teams. In nine interview sessions, a total of eleven experts were interviewed. Table 1 gives an overview of each expert's role and context. All interviewed individuals are active in industrial software engineering and occupy leading positions or have other extensive experience with agile methods in distributed settings. Even though the experts in the Alpha and Beta case are labelled as executives their companies are of moderate size and they are still actively involved in the software development process.

All interviews were held in German and lasted between 45 minutes to 75 minutes. After each interview session, the audio recording was transcribed to interview protocols and anonymised by removing individual names of people, places or company names. Two interviews were done remotely, one via telephone (Beta) and one as a video conference call (Iota), all other interviews were conducted face-to-face. The iterative data collection phase is shown in Figure 2. The

interview protocol of each interview session was used to readjust the interview guideline (like rephrasing or reordering questions and topics).

Table 1. Overview of the interviewed experts

Expert	Role	Team Size	Nr. of Sites	Countries	Process Model
Alpha	CEO and CTO	2 to 5	3 Sites	Austria	Scrum
Beta	CEO	3 to 5	4 Sites	Austria	Scrum
Gamma	Team Leader	8 to 10	2 Sites	Germany	Scrum
Delta	Two Team Leader	3 and 7	3 Sites	Austria, Slovakia	Scrum, Kanban
Epsilon	Department Manager	Varying	4 Sites	Austria, Germany	Scrum, Kanban
Zeta	Department Manager	Varying	Multiple Sites	Austria, India, Slovakia, Poland	Scrum, Kanban
Eta	Scrum Master	4 to 7	4 Sites	Austria, Germany	Scrum
Theta	Agile Coach	5 to 10	3 Sites	Austria, Hungary	Scrum
Iota	Team Leader	5 to 8	Multiple Sites	Germany & adjacent countries	Kanban

3.4 DATA ANALYSIS

To derive conclusions from the gathered data (while keeping a traceable chain of evidence) the transcripts were analysed applying the inductive category formation method adapted from Mayring [27]. This iterative process is depicted in the Data Analysis box of Figure 2.

As a starting point, some a priori codes based on the literature review and the research questions and propositions were defined and used to analyse the first interview [23]. During this analysis phase existing codes are continually adapted, new codes introduced, and existing ones renamed or merged. One code is assigned to various text passages and a text passage can also be marked with multiple codes. Beside the iterative process of adapting the codes with each analysed interview protocol, the data collection and analysis steps are again iteratively connected and were performed alongside each other [23]. The coded material is grouped into five main categories, Figure 3 shows the used codes as well as the code-category mapping.

The second category, *'Agile Practices'* examines agile practices explicitly mentioned by the teams like pair programming, continuous integration or code reviews, inquires on the implementation and if practices had to be adapted to the distributed environment. *'Communication'*, as the third main category, was one of the most prominent topics during the interviews. It investigates how communication within the team and between multiple sites is realised, which kinds of meetings exist, and which communication channels are used. The fourth category, labelled *'Distribution'*, explores general aspects of the distribution and the initial cause of the team ending up in a distributed setting. Finally, the *'Team'* section summarises team and personnel aspects like requirements for new team members and team-building aspects.
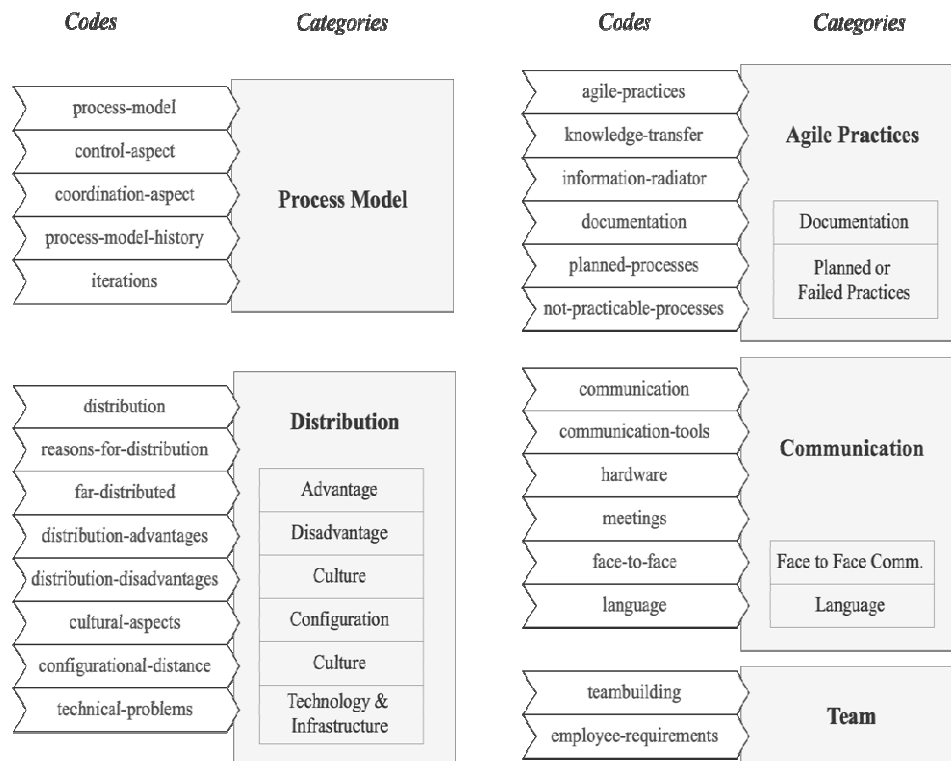
Figure 3. Codes and categories

4. RESULTS

4.1 SUMMARY

This section presents the aggregated insights from the expert interviews and is structured according to the categories illustrated in Figure 3. We worked out commonalities, point out differences between interviewed experts and present relevant quotes to underline our findings.

"There is a clear trend towards agile methods and there is a trend towards more flexible work conditions and working remote. Those trends have happened over the last ten years simultaneously. In my opinion there is no reason why you should not be able to combine the one thing with the other." (Team Leader, Iota)

4.1.1 PROCESS MODEL

Only the Iota expert favoured Kanban over Scrum in general, the other experts either applied Scrum or, as mentioned by the Delta, Epsilon and Zeta, decided individually for each project between Scrum and Kanban.

All teams execute regular planning meetings (in case of Scrum in the form of a sprint planning) as well as retrospective meetings and use short daily meetings following the standardised Scrum suggestions. The typical duration of iterations is similar to that of co-located teams and ranges from one week (Gamma) to a maximum of four weeks (Delta), all other teams apply sprints of either two or three weeks. The Iota case (applying a Kanban process) reported they do not have classical sprints, but arrange their planning and retrospective meetings similarly recurrently.

There was no straight-ahead planned path towards the respective process the teams currently apply, in all cases it was reported that the adoption was an incremental process either progressively introducing new courses of action or continuously adapting existing practices.

4.1.2 AGILE PRACTICES

All teams utilised code reviews in various forms. Alpha, Theta and Iota stated that they use pull requests as code review process and praised that method as being highly effective: Not only does it improve code quality, it furthermore has a knowledge sharing aspect because multiple team members from remote locations review code and improve it mutually. It also shifts away from single code ownership to collective ownership because every reviewer also takes up responsibility for reviewed source code. All teams reported that they intensively use continuous integration systems which also have information radiating effects. Furthermore, each team used digital boards from their project management tools to radiate information, which is not only valuable within an organisation but also for external stakeholders. Pair programming was occasionally used by most of the teams (Beta, Gamma, Delta, Epsilon, Eta, Theta) without reporting severe problems. This practice was used to transfer knowledge and information between individuals as well as locations as stated by the Beta expert:

"Pair Programming is mandatory for transferring knowledge. Because we have specialists, and everybody is specialised in certain areas [...] and we would like people to share their knowledge because if somebody leaves or becomes absent [...] another one should know what he was working on." (CEO, Beta)

> **Documentation**: Epsilon and Iota argued that documentation is more important in a distributed team than in a co-located situation because of reduced face-to-face communication and is vital to ensure knowledge transfer. One common practice most teams applied was the usage of established process management tools, the most common being Jira[1] and Confluence[2] from Atlassian (utilised by Alpha, Beta, Delta, Epsilon, Eta and Theta).

> **Planned or Failed Practices**: When teams came up with new suggestions, they tried them in the following iterations rather than planning long ahead. Therefore, no practices were mentioned that were not yet applied but planned for the future. One exception was the Theta expert who reported that the team changed longer meetings like sprint planning and retrospectives away from a complete remote approach towards meeting regularly in one location:

> *"In the past, I tried doing planning sessions remotely, but the costs were not justifiable. You just have so much overhead in communication, so many misunderstandings and call-back inquiries that the efficiency cannot be compared to being co-located. Just such simple tasks like going through the backlog can take up three times as much time as when doing it face-to-face."* (Agile Coach, Theta)

4.1.3 COMMUNICATION

Every team had some tool or setup serving as a primary communication channel. Most teams used Skype (Beta, Delta, Epsilon, Zeta, Eta and Theta), while two teams shifted to a tool called Slack (Alpha and Iota). Meetings are done with common audio and video communication tools. Zeta and Theta reported that they have special meeting rooms for team meetings with dedicated

hardware like microphones, large screens or LCD projectors. There was no consent about the usage of video conferencing, some use audio as well as video channels while other teams just use audio channels for certain meetings.

Face-to-Face Communication: The first important reason for regular face-to-face meetings is to build trust and interpersonal relationships between team members. During the distributed collaboration, communication mainly focuses on professional aspects and not personal matters and a lot of informal communication situations (like lunch, coffee breaks or after-work activities) are missing. Secondly, when it comes to complex topics (like retrospectives or long planning session) or a lot of attendees, such situations are preferably done co-located.

"Especially there [at the retrospective meetings] it is important to have all people in one location. The discussions are taking longer, you need a whiteboard from time to time to sketch something and explain something to others. This is very difficult when doing it remotely; it does not work." (Team Leader, Iota)

Language: Even in teams within the same country or distributed within adjacent ones, having multiple native tongues within a team can raise the complexity of communication. Gamma, Delta, Theta and Iota explicitly mentioned this factor and that it should not be underestimated.

4.1.4 DISTRIBUTION

The main reason for teams ending up distributed was due to human resource aspects: needed know-how or resources were not available in one location and therefore it was necessary to add remote team members.

Advantages: Having access to more employable candidates and being able to choose from a broader range of skilled individuals was named as one of the main benefits. Furthermore, teams are more flexible because their processes and tools are set up for remote work giving team members more freedom in choosing their workplace. A third aspect (found in cases Zeta and Eta) was the ability to be near to customers and generally being present in multiple locations as a company and thus increase visibility.

Disadvantages: Team members are more detached from each other which impairs communication. Finding the right communication tools for the team can be time-consuming and remote communication setups like conference rooms and hardware have to be serviced and maintained. Generally, this additional effort adds complexity which increases potential error sources in the future. Other reported issues (by Beta, Gamma, Epsilon and Zeta) are the increased costs and time travelling entails.

Culture: Despite the low geographical distance, some teams had colleagues with different nationalities and cultural backgrounds. All of them reported that this diversity is having a positive effect due to different approaches and mindsets. The only cultural aspect that was reported to be challenging was different spoken languages and the communication problems those can bring if not accounted adequately.

Configuration: Configurational distance was mentioned explicitly by Alpha and Theta but was also hinted in other cases. Alpha reported the bad experience single remote team members joining a meeting where a larger group of co-located sometimes had when the co-located group did not pay enough attention to maintain proper conduct. Theta similarly stated

that imbalance could be especially problematic in meeting sessions where the majority of the team is co-located in one place and just a few individuals are attending remotely.

Technology and Infrastructure: More than half of the interviewees reported bad infrastructure and internet availability as one of their biggest problems.

4.1.5 TEAM

All experts applied team building strategies like bringing the team together on a regular basis for team events. Experts agreed that it is necessary for team members to be communicative and self-organising and that a communicative and trusting team is an essential condition for successful remote collaboration.

4.2 RESEARCH PROPOSITIONS REVISITED

P1: Iterations in distributed agile teams are similar in length to iterations in co-located teams.

All teams stayed within the typical maximal sprint duration of 30 days (which was applied just by Delta). Apart from Gamma (one-week sprints), seven out of the nine interviewed teams had iteration durations of two or three weeks. Therefore, the empirical data supports this proposition in a low distributed setting.

P2: Modern software and tools are a major factor for the success of distributed teams.

All teams used software tools designed for agile project management that provide virtual boards and visual representations used to spread and share information between locations functioning as an information radiator. Furthermore, they offer a regular channel every team member can use at any time which is without much obligation and therefore reduce the perceived distance.

"Regarding the tools, we have available to support our mode of operation I do not see a problem. The available selection you can pick from is huge and they are still constantly improving. There for sure is room to still upgrade and improve but for my daily routine there are very few challenges that have not already been solved." (Team Leader, Iota)

P3: Aside from communicative skills there are no special requirements for team members in distributed teams.

No team named any essential technical requirements, but all stated that the ability to communicate is a vital skill for every team member. Except for two, the experts stated that junior level experience is sufficient to collaborate successfully in a distributed team. On the other hand, Beta and Gamma argued that remote team members should be generally senior level because they felt the training and on-boarding of inexperienced developers is more difficult due to the communication impediments of distance.

P4: While for short, standardised meeting situations remote communication is sufficient, face-to-face communication is preferred in longer, informal situations.

Daily meetings and general meetings of short duration (with durations typically not planned much longer than one hour) are done smoothly with audio or video conferencing. When it comes to informal, complex communication situations on the other hand, all interviewees reported that they prefer to bring their teams together in one physical location if possible.

P5: Technical faults and limitations pose a serious issue for distributed communication.

Infrastructure was an issue that was pointed out in multiple situations during the interview. Alpha, Beta, Delta, Epsilon and Iota explicitly mentioned bad internet connectivity and infrastructure as a significant downside of the distribution due to the negative impacts on communication. Eta (and Epsilon) furthermore pointed out the dire situation during travel:

"In transit [between the two offices] I do not join any meeting. Maybe briefly if somebody calls me - but generally, you know that on this route the network connection fails two or three times, you cannot really participate in a meeting." (Scrum Master, Eta)

P6: Informal and frequent communication improves collaboration between remote colleagues.

As mentioned in Section 4.1.3 all teams had primary communication channels like Skype or Slack. In 5 out of the 9 interviews experts mentioned the benefits of such a channel: That it creates proximity and a feeling of sitting together and therefore reduces the perceived distance between team members. One advocate of such a channel was the Epsilon expert who argued:

"A big amount of communication [...] is done via a chat system. A classic chat system where you can join easily, other people respond quickly, and you have an ongoing communication flow for the whole day. This works because it can be used casually alongside the work." (Department Manager, Epsilon)

4.3 RESEARCH QUESTIONS REVISITED

RQ: How are agile processes used in a distributed team (limited to a low geographical and time dispersion)?

When applying agile methods like Scrum or Kanban, the overall process, applied practices, iteration durations and team sizes are generally equal to co-located teams. The big difference, the geographical distance preventing face-to-face communication and direct interaction, is met by replacing this direct communication channel by various digital alternatives like text chats, audio calls or video conferences. Interacting with remote colleagues is done with the excessive use of collaboration and management tools which function as information radiators to keep distant team members informed. Although no substantial barrier, there are aspects that require increased attention like the scheduling of meetings or extra effort to uphold communication between sites.

"In my opinion, this [the constraint of co-location] has changed by now, people are noticing and hearing things because they are constantly chatting with each other, that is similar to talking. Maybe it is even better because it does not disturb you, you can inquire when you want and are not forced to listen at a certain moment. That is definitely an advantage." (Department Manager, Epsilon)

RQ.A: Which challenges have to be faced using agile methods and how can those issues be handled?

This section investigates the impact and significance of the three main challenges of distribution, as well as other impediments that were identified.

> **Communication**: Out of the three major challenges discussed in Section 2.2, communication is arguably the biggest hindrance in remote agile teams. Agile methods strongly rely on frequent communication and trust within a team. Although not an

insuperable barrier it still requires attention and strategies to improve the situation due to the crucial influence communication has on coordination and control. Regular face-to-face contact is used to build and foster a team spirit and interpersonal relationships. All experts reported that they periodically bring together the whole team to deal with complex topics like architecture planning or retrospective meetings. The Eta expert argued explained their approach like this:

"What is important if I do not have the optimum like being co-located in one big room is to try and solve the problems another way. We, for example, say: If we are distributed, we do at least the sprint transitions in the same location." (Scrum Master, Eta)

Language: Team members who do not share the same native tongue were quite common despite the focus on low geographical distance and such a case was directly mentioned in four out of nine interview sessions. Remote communication lacks various channels, making it especially hard for non-native speakers to follow conversations and to argue with colleagues [28]. Different language levels therefore are having a bigger impact than in co-located situations. To account for this challenge, raising awareness about this obstacle and the different aspects of remote communication should be the first step. Furthermore, improving communication possibilities by providing proper software as well as hardware to improve the quality of the transmitted audio and video channels.

Awareness: Creating and maintaining awareness about the status of remote team members can be challenging. One team leader of the Delta case stated: "What we miss are some basic things like knowing if your teammate is still at work or already at home. If you are in the same room, you can take a look at his place or ask a co-worker" (Team Leader 1, Delta). Alpha and Beta use the status setting of their communication tools (Slack and Skype) to display the current state of team members: There are various agreed upon states like *'at work'*, *'in a meeting'* or *'on holiday'* which are used to tell the current status to remote colleagues. Teams have to understand and accept that collaboration in distributed teams differs from co-located ones and that it is necessary to apply practices (like frequent meetings) and tools to counter lacking awareness of remote occurrences and events.

Technology and connectivity: Communication and collaboration in distributed teams build on reliable internet connectivity technology like communication hard- and software. A regularly reported issue is the lack of those requirements like an unsteady network connection which massively impedes synchronous and frequent communication, in turn having a negative impact on coordination and control. One measure to reduce such obstacles is to invest in quality hardware and stable internet- and wireless connections. Having backup plans and protocols in case of failing communication channels also can reduce disconcertment within a remote team.

More detailed solutions and proposals to overcome those challenges and avoid pitfalls are described in Section 5.1.

RQ.B: Which benefits result from pursuing agile methods in a distributed setting for a team?

Agile methods have an especially positive impact on control and coordination challenges. In detail the following benefits on geographically distributed teams were identified:

Compensation of coordination and control challenges: These challenges are mitigated by short iterations and fast feedback loops of agile methods. While setting up clear

schedules and practices are aspects of formal control, informal control is induced by frequent meetings and the self-organising aspect of agile teams. Short iterations and daily meetings are two aspects of agile methods that alleviate issues of coordination. Combined with modern project management tools and digital boards serving as information radiators the coordination challenge is strongly mitigated.

"Agile methods make it easier because I am faster in control. Due to the daily standup and the fact that upcoming work is broken down to small work item pieces I am able to see progress much faster." (Department Manager, Zeta)

Team spirit and collaboration: Agile methods build on frequent communication which helps teams accept the geographical distance and try to overcome it. A high amount of communication within a team positively impacts collaboration between geographical locations.

A Quicker revelation of communication and collaboration issues: Obstacles like misapprehensions can lead to problematic outcomes in distributed teams but frequent communication, short iterations and feedback loops reveal obstacles very swiftly.

"The reason why I think agile methods are very helpful in distributed teams is that they generally focus on open and short-term communication, communication in short iterations. What I have seen in distributed teams without short communication iterations is that they drift apart very easily." (Agile Coach, Theta)

Spreading of information between sites: They are excellent ways of radiating information: Technical aspects like automated tests and continuous integration systems constantly monitor the status and inform team members about a project's status. Practices like pair programming or daily meetings furthermore regularly update all team members about progress and directions of a project.

4.4 LIMITATIONS

We achieved internal validity by relying on theoretical propositions as a starting point. Furthermore, the data is gathered from multiple different teams, grouped and compared to each other. Regarding external validity targeting the generalisability of results, our narrow focus to low distance does not allow generalisation beyond that specific context. Especially insights regarding the infrastructure situation are very specific to certain locations. The 'Temporal Distance' dimension was not analysed in accordance with our focus listed in Section 3.1 and because all investigated teams were within the same time zone.

The proposed recommendations listed in Section 5.1 are derived from the analysed data and are yet to be validated in future work.

4.5 COMPARISON TO RELATED WORK

Zieris and Salinger [29] investigated a very similar agile near shoring setting, where two polish Scrum teams worked remotely for a German company. They summarised that agile near shoring "is feasible and may produce high satisfaction amongst Product Owners" which corresponds with our conclusion. They furthermore highlighted the importance of awareness about agile practices as well as the distance between team members, two essential aspects similar to our findings. The researched teams applied an extra meeting ahead of the daily scrum they called "Synchronization Meeting" which had no strict rules: "The whole meeting is driven by the developers' needs and continues until no more questions pop up. Hence, there is no time-boxing: If need be, the Sync

will be resumed right after the Daily Scrum, practically occupying all developers". This practice resembles closely to our fourth recommendation ("Plan for communication") in Section 5.1.

Wawryk et al. [30] performed a case study about a Scrum team which started as a co-located team in Austria and then was scaled with colleagues from Germany and Bosnia. They identified the language as one main challenge faced by the distributed team.

Furthermore, they reported that there were no major cultural differences because of cultural awareness of team members. In contrary according to the teams' feedback, "all team members were open and curious about other cultures and lifestyles". What constituted a challenge were the different work time models and holidays of the different countries. One of the biggest challenges was the high need for communication. Another aspect which resembles our findings (and is one of our recommendations in Section 5.1.) was the positive aspect of team building activities where the whole team was co-located in one location.

Generally, agile methods are beneficial in distributed teams [31], but it is a notable aspect that teams distributed within a limited distance still face similar challenges like globally distributed ones. A systemic review from Alzoubi et al. [32] investigating communication challenges in geographically distributed agile development (GDAD) and techniques to overcome those challenges on a global distributed scale. Some of their identified techniques to overcome communication challenges resemble our recommendations in Section 5.1. To overcome distance differences they recommend to "encourage regular visits and face-to-face communication, create a structure of trust, minimize dependencies among teams, encourage individuals to work closely with both developers and project management teams, enforce meetings and commitment, localize component ownership, enhance coordination by promoting social skills" or in terms of tooling they suggest to "switch to the most appropriate tools" and "use synchronous and asynchronous tools".

5. CONCLUSIONS

The progress of technology and evolution in collaboration tools increased the number of software engineers working in distributed teams year by year. Our results show that applying agile methods in distributed teams with low geographical distance poses no problem but rather brings forth several benefits. Common agile practices were mastered successfully by replacing face-to-face communication with a variety of digital communication channels. These practices not only improve the software engineering process but furthermore pose additional communication channels and information radiators which in turn improve collaboration.

There were no deviations to basic agile principles like team size and iteration length compared to co-located teams. Furthermore, there were no agile practices that were doomed to fail due to the distributed setting, instead every team managed to apply all the techniques and processes they felt worth using.

Short iterations and frequent and encouraged communication of agile methods are solving problems of coordination and control and no investigated case reported any serious obstacles in those areas. Iteratively and constantly reviewing, re-evaluating and re-prioritising the workflow and work items increases social cohesion in distributed teams. Procedures like short sprint cycles and frequent short meetings like daily standups, plannings and retrospectives were mentioned as adjuvant reasons.

The importance of communication was significant, and the topic came up in various situations and forms during the interviews. Modern technology in the form of a wide variety of text, audio or video communication tools, is nowadays providing enough flexibility and convenience to

create a constant stream of communication between remote team members. The biggest identified obstacles are located in the areas of communication. One aspect is infrastructural issues like unreliable tools, hardware or unstable internet connections. Another surprising result was that while cultural differences posed no problem at all but were on the contrary welcomed by all experts, different native languages and unequal language abilities are possible pitfalls that reportedly can impede communication between remote team members.

An essential distinction between far and near distributed teams is the fact that the teams investigated in this paper had the characteristic of being able to gather team members in one place within reasonable costs. Although all teams were generally set up for remote collaboration, they still chose to co-locate the whole team in one location regularly. Such collective situations are used to deal with complex topics like architecture planning or retrospective meetings. Furthermore, regular face-to-face contact builds and fosters a team spirit and interpersonal relationships.

5.1 RECOMMENDATIONS

The discussion of the research propositions and answering the research questions in the previous section summarised four challenges and four benefits of agile methods in distributed teams with low spatial and temporal distance. Based on those findings the authors make the following seven proposals derived from the examined teams to improve the effectiveness of distributed agile teams further:

#1 Use practices and tools to radiate information: Various agile practices are well suited to not only produce source code but also to foster communication and spread information. Tools like issue trackers or digital boards furthermore can serve as information radiators. Pair programming is a valuable practice to share knowledge between different locations when remote team colleagues collaborate. Text-based code reviews or the usage of pull-requests are practices that as a side effect communicate information and can update colleagues about progress.

#2 Co-locate the team regularly: Collective activities when being co-located are necessary to build and maintain a successful team. The interviewed experts therefore all mentioned the importance of bringing the team together frequently. Especially within the boundaries of a low geographical distance this is possible within reasonable time and costs.

#3 Provide channels and encourage communication: Teams need to have the possibility to communicate whenever they want, getting in touch with a remote colleague has to be as easy as possible. Therefore, good communication infrastructure like microphones, speakers, TV-screens or headphones as well as software tools is crucial.

"That [bad quality of remote communication tools] is the reason why infrastructure is so important and is an aspect where you must not skimp. Also, provide the team with multiple communication channels so they can choose which suits them best." (Scrum Master, Eta)

#4 Plan for communication: Spreading information in distributed teams does not happen as naturally as in co-located situations, informal and personal communication is often neglected. Therefore, it is advisable to schedule dedicated communication opportunities and meetings. This also applies to more informal topics, Theta, for example, reported they once clinked glasses virtually in a video conference. Virtual activities like shared coffee breaks or collective lunchtime over video chats can increase otherwise lacking informal communication.

#5 Have alternative communication strategies prepared: When distributed teams communicate via audio or video channels these channels are error-prone, a fragile internet

connection, software or hardware defects are impeding communication. Therefore, it is advisable to have backup routines like for example usage of text-based channels, alternative internet access, telephone lines and general standby hardware.

#6 Adapt the process model: Most agile process models are intended for co-located teams and build on direct face-to-face communication. Therefore, it is necessary to adapt processes and practices to a distributed situation and also regularly review and improve them. One example would be doing practices that normally are done in a synchronous communication situation asynchronously (like asynchronously performed standup meetings in a text chat or code reviews that are discussed via text comments).

#7 Stick to the defined processes and practices: Team members should be able to rely on established processes. When a reoccurring meeting is scheduled at a specific date, this date should be steady. Re-scheduling on short notice may be more problematic since all remote team members have to be notified.

5.2 FUTURE WORK

In 2016, around 25 per cent of German software engineers have experience with working at least part-time remotely [2] and there is no indication that this number will decrease anytime soon. Due to the nature of empirical qualitative research, our resulting arguments and findings within the defined boundaries could be evaluated in a quantitative study to gather further insights and also strengthen the findings or suggest alterations. Our seven recommendations could be subject to follow-up studies focusing on their in-depth evaluation based on larger empirical data sets.

REFERENCES

[1] M. Kajko-Mattsson, G. Azizyan, and M. K. Magarian, "Classes of Distributed Agile Development Problems," in Proceedings of the 2010 Agile Conference, Washington, DC, USA, 2010, pp. 51–58.

[2] "Stack Overflow Developer Survey 2016 Results," Stack Overflow. [Online]. Available: https://insights.stackoverflow.com/survey/2016. [Accessed: 08-Nov-2018].

[3] P. J. Ågerfalk, B. Fitzgerald, H. Holmström, B. Lings, B. Lundell, and E. Ó. Conchúir, "A framework for considering opportunities and threats in distributed software development," in In Proceedings of the International Workshop on Distributed Software Development (Paris, Aug. 29, 2005). Austrian Computer Society, pp. 47–61.

[4] Ebert, Christof, "The Bright Side: Benefits," in Global Software and IT, Wiley-Blackwell, 2011, pp. 15–18.

[5] J. Kotlarsky and I. Oshri, "Country attractiveness for offshoring and offshore outsourcing: Additional considerations," JIT, vol. 23, pp. 228–231, Dec. 2008.

[6] J. Noll, S. Beecham, and I. Richardson, "Global Software Development and Collaboration: Barriers and Solutions," ACM Inroads, vol. 1, no. 3, pp. 66–78, Sep. 2011.

[7] E. Woodward, S. Surdek, and M. Ganis, A Practical Guide to Distributed Scrum, 1st ed. IBM Press, 2010.

[8] K. Beck and C. Andres, Extreme Programming Explained: Embrace Change, 2nd Edition, 2nd edition. Boston, MA: Addison-Wesley, 2004.

[9] J. Shore and S. Warden, The Art of Agile Development, First. O'Reilly, 2007.

[10] K. Beck et al., Manifesto for Agile Software Development. Online available at https://agilemanifesto.org/; last access 01.12.2018), 2001.

[11] M. Cohn, Succeeding with Agile: Software Development Using Scrum, 1 edition. Upper Saddle River, NJ: Addison-Wesley Professional, 2009.

[12] T. J. Allen and G. Henn, The Organization and Architecture of Innovation: Managing the Flow of Technology, 1 edition. Amsterdam; Boston: Butterworth-Heinemann, 2006.

[13] V. Casey, "Imparting the Importance of Culture to Global Software Development," ACM Inroads, vol. 1, no. 3, pp. 51–57, Sep. 2011.

[14] J. D. Herbsleb, "Global Software Engineering: The Future of Socio-technical Coordination," in 2007 Future of Software Engineering, Washington, DC, USA, 2007, pp. 188–198.

[15] M. B. O'Leary and J. N. Cummings, "The Spatial, Temporal, and Configurational Characteristics of Geographic Dispersion in Teams," MIS Q, vol. 31, no. 3, pp. 433–452, Sep. 2007.

[16] E. Carmel and R. Agarwal, "Tactical approaches for alleviating distance in global software development," IEEE Softw., vol. 18, no. 2, pp. 22–29, Apr. 2001.

[17] L. Lee-Kelley and T. Sankey, "Global virtual teams for value creation and project success: A case study," Int. J. Proj. Manag., vol. 26, no. 1, pp. 51–62, Jan. 2008.

[18] J. R. Evaristo, R. Scudder, K. C. Desouza, and O. Sato, "A dimensional analysis of geographically distributed project teams: a case study," J. Eng. Technol. Manag., vol. 21, no. 3, pp. 175–189, Sep. 2004.

[19] G. O. Wiredu, "A Framework for the Analysis of Coordination in Global Software Development," in Proceedings of the 2006 International Workshop on Global Software Development for the Practitioner, New York, NY, USA, 2006, pp. 38–44.

[20] J. E. Moore, C. K. Williams, and M. Sumner, "The Role of Informal Control in PMO Lite Environments," in Proceedings of the 50th Annual Conference on Computers and People Research, New York, NY, USA, 2012, pp. 27–30.

[21] J. Ritchie, J. Lewis, C. M. Nicholls, and R. Ormston, Qualitative Research Practice: A Guide for Social Science Students and Researchers. SAGE, 2013.

[22] D. Šmite, C. Wohlin, Z. Galviņa, and R. Prikladnicki, "An Empirically Based Terminology and Taxonomy for Global Software Engineering," Empir. Softw Engg, vol. 19, no. 1, pp. 105–153, Feb. 2014.

[23] P. Runeson, M. Host, A. Rainer, and B. Regnell, Case Study Research in Software Engineering: Guidelines and Examples, 1st ed. Wiley Publishing, 2012.

[24] R. K. Yin, Case Study Research, 5th ed. Los Angeles: Sage Publications, 2013.

[25] L. Layman, L. Williams, D. Damian, and H. Bures, "Essential communication practices for Extreme Programming in a global software development team," Inf. Softw. Technol., vol. 48, no. 9, pp. 781–794, Sep. 2006.

[26] S. Dorairaj, J. Noble, and P. Malik, "Effective Communication in Distributed Agile Software Development Teams," in Agile Processes in Software Engineering and Extreme Programming, 2011, pp. 102–116.

[27] P. Mayring, Qualitative content analysis: theoretical foundation, basic procedures and software solution. 2014.

[28] A. Cockburn, Agile Software Development: The Cooperative Game, 2 edition. Upper Saddle River, NJ: Addison-Wesley Professional, 2006.

[29] F. Zieris and S. Salinger, "Doing Scrum Rather Than Being Agile: A Case Study on Actual Nearshoring Practices," presented at the Proceedings - IEEE 8th International Conference on Global Software Engineering, ICGSE 2013, 2013, pp. 144–153.

[30] V. J. Wawryk, C. Krenn, and T. Dietinger, "Scaling a running agile fix-bid project with near shoring: Theory vs. reality and (best) practice," in 2015 IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 2015, pp. 1–7.

[31] H. Holmström, B. Fitzgerald, P. J. Ågerfalk, and E. Ó. Conchúir, "Agile Practices Reduce Distance in Global Software Development," Inf. Syst. Manag., vol. 23, no. 3, pp. 7–18, Jun. 2006.

[32] Y. I. Alzoubi, A. Q. Gill, and A. Al-Ani, "Empirical studies of geographically distributed agile development communication challenges: A systematic review," Inf. Manage., vol. 53, no. 1, pp. 22–37, Jan. 2016.

AUTHORS

Manuel Stadler received his master's degree in software engineering at the Vienna University of Technology in 2016. His research interests focus on software engineering processes and distributed collaboration in software development teams. He has several years of IT industry experience as a Software Engineer as well as Project- and Product Manager.

Raoul Vallon is a postdoctoral researcher at Vienna University of Technology, where he leads the AMMA (Amazing Makers) work group on empirical software engineering at the Research Group for Industrial Software (INSO). In 2014 he was a visiting researcher at Professor Larry Leifer's Center for Design Research at Stanford University. Finishing in 2016, he developed the ADAPT (Agile Distributed Adaptable Process Toolkit) framework as part of his PhD research. He also has several years experience working in industry as a software engineer and manager.

Martin Pazderka is a Senior Lecturer at Research Group for Industrial Software (INSO) at the Vienna University of Technology. His research interests are project management, organisational maturity models, and social dynamics in software engineering teams. He has 20+ years of IT industry experience in various roles ranging from IT-Architect to Project and Program Manager, has successfully delivered complex international projects for several Fortune 500 clients, and contributed to PMI's OPM3 standard.

Thomas Grechenig is a senior architect in large IT systems and nation-wide IT-infrastructures. He is a professor for industrial software engineering and the head of the Research Group for Industrial Software (INSO) at the Vienna University of Technology. He and his teams have planned, designed, and built several large-scale IT solutions, e.g. in health, railway systems or public transport applications. In science and research the focus of interest goes towards enhancing the stability and fine-tuning of the IT-mass-concept in all its critical sectors (usability, security, IT-infrastructure, performance, integration and interoperability).