



ISSN 1943-7544

Research Paper

doi: 10.17705/1pais.15202

Volume 15, Issue 2 (2023)

# Pacific Asia Journal of the Association for Information Systems

## Interaction Between Agile Methods and Organizational Culture – A Qualitative Study

Marja Känsälä<sup>1\*</sup>, Norio Tokumaru<sup>2</sup>

<sup>1</sup>Finnish Institute of Occupational Health, Finland, [marja.kansala@ttl.fi](mailto:marja.kansala@ttl.fi)

<sup>2</sup>Kansai University, Japan, [norio-t@kansai-u.ac.jp](mailto:norio-t@kansai-u.ac.jp)

### Abstract

**Background:** *Agile practices have gained popularity in the 21st century. There is also a growing body of research on agile methods. While some earlier research on agile practices and organizational culture exists with the assumption that the optimal combination of agile practices and organizational culture exists, we examine how agile methods and organizational culture interact and are mutually adjusted.*

**Method:** *To find out how agile methods and organizational culture are related in practice, we conducted 50 semi-structured interviews with software and embedded software developers from five Japanese companies and three software teams from Finland.*

**Results:** *Adopting agile methods does not necessarily cause convergence of organizational culture toward democratic which seems to be most compatible with archetypal agile practices. Agile methods can be adapted to organizational cultures that are sometimes challenging to modify. We demonstrate that companies tried to fit agile practices into hierarchical organizational cultures, demonstrating that organizational culture can occasionally be viewed as an exogenous variable influenced firm-, product-, and industry-specific features. Also, it is possible to transform organizational cultures to democratic. The examples demonstrate how companies alter their culture in part through HRM techniques. It should be noted, however, that these efforts to conform to culture can have some restrictions.*

**Conclusion:** *There is not a single ideal combination of agile methods and organizational culture, despite what some earlier studies suggested. The results show how organizational culture and agile methods interact and adapt to one another in different ways. According to several earlier studies, different organizational cultures may support different facets of agile methods. They do not presuppose organizational culture diversity; in that it goes beyond the notion of a one-to-one relationship between agile methods and democratic organizational culture. While implementing agile practices in various organizational cultural contexts, practitioners must consider how agile methods and organizational culture are interconnected.*

**Keywords:** Agile Methods, Content Analysis, Organizational Culture, Qualitative Research, Semi-Structured Interviews.

This research article was submitted on November-2022 and under two revisions, accepted on April-2023.

Citation: Känsälä, M., & Tokumaru, N. (2023). Interaction between Agile Methods and Organizational Culture – A Qualitative Study. *Pacific Asia Journal of the Association for Information Systems*, 15(2), 32-59. <https://doi.org/10.17705/1pais.15202>

Copyright © Association for Information Systems.

## Introduction

Agile methods have become popular in software development. They were developed to enable systematic project management even when perfect planning is not possible in the beginning stages of a project. Agile methodologies originate from software process improvement and have roots in lean methods in industrial R&D (Takeuchi & Nonaka, 1986). Agile practices are realized through various methodologies, such as Scrum and Kanban, including general project management guidelines and technical practices. Communication and coordination methods within agile include co-located teams and daily meetings (Kakar, 2017), however, there are variations within the methods as well. For instance, Kanban permits specialized teams while Scrum recommends cross-functional teams (Kniberg & Skarin, 2010).

While agile methods have mostly been studied from the perspective of software functionality and productivity, this article focuses on the human and social aspects, and within those aspects, more specifically organizational culture. Organizational issues have previously only been briefly addressed in the software engineering literature, but human and social factors have become a thematic group of agile literature (e.g., Lalsing et al., 2012). The Agile Manifesto (n.d.) (<https://agilemanifesto.org>) divides the four agile values into 12 principles (Agile Manifesto Principles, n.d., <https://agilemanifesto.org/principles.html>). Perhaps the most significant value associated with the human side of agile methods is that of valuing individuals and interactions over processes and tools. Of all the principles, perhaps some of the most important ones in relation to the human and social issues are: 1) Build projects around motivated individuals, give them the environment and support they need, and trust them to get the work done; and 2) Best results emerge from self-organized teams.

Since successful agile adoption requires major organizational and process change, it is crucial to first concentrate on the related social and cultural concerns (Gren et al., 2017). This study examines how organizational culture and agile methods interact and are mutually adjusted. We stress this interactive part because it has been researched how organizational culture and agile adoption are related (e.g., Tolfo & Wazlawick, 2008). We conducted 50 semi-structured interviews with software and embedded software developers in five companies and three software teams in one company. This allows a comparison setting of agile methods and organizational culture. The structure of this article is as follows. First, we review literature in terms of agile methods and organizational culture. Secondly, we discuss materials and methods, including validity and reliability. Thirdly, we present our results. Then we discuss our results, limitations and ideas for further research. Finally, we present the conclusions.

## Literature Review

### *Agile Methods*

There are many studies highlighting the opportunities provided by agile methods. Agile methods have introduced new ways of organizing teamwork and shared leadership. The methods greatly favor teamwork over individual role assignment (Annosi et al., 2016). Agile practices, for instance, were found to be more flexible and empowering than other management techniques that placed more emphasis on management control in Hogdson and Briand's (2013) study. Commitment to teamwork is one indicator of agility (Gren et al., 2017). Agile approaches have been linked to team morale and empowerment in previous studies (Gren et al., 2017; Senapathi & Srinivasan, 2012). Team experience, attitude, and competence have been identified as success factors for the assimilation of agile in organizations (Senapathi & Srinivasan, 2012). Although there are friction points and some fundamental aspects of agile approaches that seem incompatible with the restrictions placed

by regulated environments, agile methods can be successfully deployed in a regulated environment (Fitzgerald et al., 2013).

Although historical focus has been on opportunities related to agile practices, this view has been shifting to the risks (e.g., Cram, 2019). If these risks are not considered when applying agile practices, the methods may challenge teamwork. The implementation of agile techniques may occasionally be perceived by developers as an attempt to micromanage (Marchenko & Abrahamsson, 2008). In pseudo agile environments, metrics are often perceived as a threat, make people uncomfortable, and can do more harm than good (Liechti et al., 2017). Self-intensification, for instance, could happen if these problems are not considered when using an agile type of work (Drury et al., 2012; Porschen, 2012). Lack of empowerment to make decisions and dependency on leaders for decisions despite team autonomy are frequently problematic aspects of agile team decision-making (Drury et al., 2012; Lazwanthi et al., 2016). Agile teams should no longer rely on the project manager to manage requirements in the context of self-organizing teams (Hoda & Murugesan, 2016). An imbalanced workload for the Scrum Master might result from a lack of self-organization (Jovanović et al., 2017). Insufficient transition to new responsibilities and concerns with the project's integration at the organizational level are other problems that are preventing the adoption of agile (Hoda & Murugesan, 2016; Jovanović et al., 2017). Rigid organizational structures can be problematic, and intra-organizational cooperation has been identified as a crucial aspect of agility (Yauch & Adkins, 2004).

### **Organizational Culture**

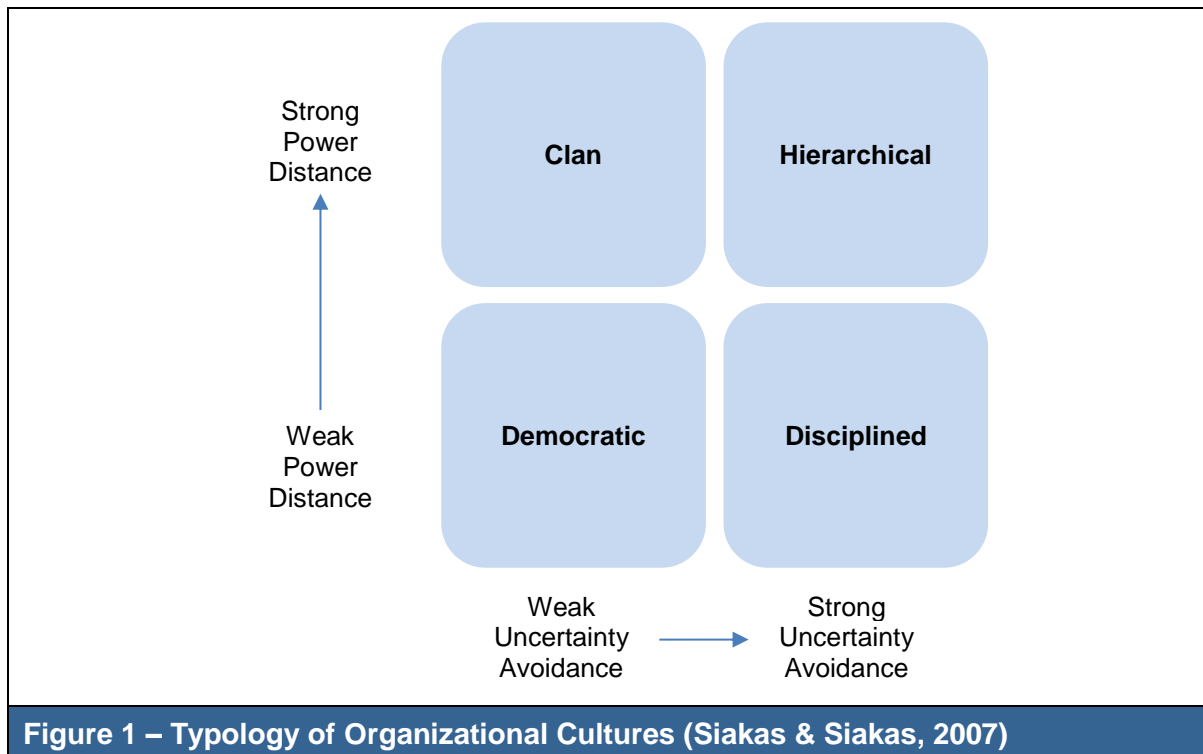
Schein (2010) states that there are layers within the organizational culture: artefacts, values and norms, and basic assumptions. Siakas and Siakas (2007) suggest that organizational culture can be studied, for example, through visible organizational artefacts, structures, and processes, as well as the level of practices. Schein is often criticized for presenting organizational culture as non-changing and monolithic. The criticism contends that organizational culture ought to be viewed as more dynamic, continuously changing, and local. Four types of organizational forms are identified by one typology: clans, hierarchies, markets, and networks (Seppänen & Poutanen, 2020; Spinuzzi, 2015). The competing values model (e.g., livari & Huisman, 2007) divides organizational culture into four categories: group, developmental, rational, and hierarchical. Hyun et al. (2020) distinguishes between collectivistic culture, which emphasizes conformity and communal behavior among members, and democratization culture, which values the willingness to share information and the acceptance of diversity.

To reach consensus and agreement and prevent cultural confrontations within a unit as well as within the organization, it is crucial to have a consistent culture (Siakas & Siakas, 2007). People manage and resolve difficulties brought on by the coexistence of organizational types such as clan and hierarchy in their practices (Seppänen & Poutanen, 2020; Spinuzzi, 2015). In a case study of an engineering company's hardware and software groups, different clan cultures developed within a bureaucratic hierarchy as a clannish group of engineers and hierarchical bureaucracy clashed over workplace culture (Artemeva & Freedman, 2001; Spinuzzi, 2015). Cultural differences between hardware and software engineers caused them to identify more strongly in clans and the tensions between hierarchy and clan configurations resulted in internal contradictions. It is also possible to find coherence, ways to reconcile the internal contradictions to work within all configurations (Spinuzzi, 2015). Hyun et al. (2020) finds that consistent organizational culture allows successful adaptation to a changing environment.

## **Agile Methods and Organizational Culture**

There is some prior research on organizational culture and agile practices (e.g., Hyun et al., 2020; MacGregor et al., 2005; Tolfo et al., 2011). The body of existing literature underlines how crucial company culture is to the success of implementing agile methods. To successfully deploy agile practices, it is critical to comprehend how they interact with organizational culture (Lazwanthi et al., 2016). Tolfo et al. (2011) find that identifying and understanding cultural aspects of the company are essential when adopting agile methods, and Hussman (2004) identifies promoting and maintaining an adequate culture as the key to the success of any agile project. Inadequate culture will make it difficult to become agile (Tolfo et al., 2011). Gupta et al. (2019) examined the relationship between IT department culture and agile practices. They also find that culture can be a potentially important barrier to successful implementation of agile methods and antagonist to agility. Poth et al. (2019) find that applying agile methods is challenging in traditional as opposed to agile environments. The existing literature of agile methods and organizational culture has identified typologies of cultures and evaluated which ones are the most compatible with agile methods. As an example of categorization of organization cultures within agile methods power, cooperation, and bureaucratic cultures are recognized by Lazwanthi et al. (2016) and Tolfo et al. (2011). Applying agile methods can be harder in power and bureaucratic cultures (Lazwanthi et al., 2016). Shared assumptions between different units form organizational culture and different sub-cultures will appear for each unit inside the organization within agile methods (Tolfo et al., 2011).

In their research of agile methods, Siakas and Siakas (2007) suggest a typology of organizational cultures, including clan, hierarchical, democratic, and discipline, based on Hofstede (1980, 2011) (See Figure 1). In contrast to Hofstede's (1980, 2011) initial work, which focused on national cultures, we use the dimensions as Siakas and Siakas (2007) in this study to examine organizational culture. Power distance (PD) is on the vertical axis, and uncertainty avoidance is on the horizontal axis (UA). The power distance index characterizes how accepting and anticipating less powerful members of organizations are that power is not distributed unequally (Hofstede Insights, n.d.). Being independent; coaching, facilitating, and empowering management; leadership relying on the experience of team members; and participative communication are characteristics of weak power-distance cultures. When a culture's members feel threatened by ambiguous or unknowable events, they develop beliefs and institutions that aim to avoid it, which is referred to as uncertainty avoidance (Hofstede Insights, n.d.). Need for rules, risk evaluation before the start of a project, and managers asking for detailed information before making decision are examples of practical expressions of uncertainty avoidance. We choose to adopt this classification because we believe that the power distance and uncertainty avoidance dimensions accurately capture the core of agile values and principles: The dimension of power distance reflects the agile principles of building projects around motivated individuals and trusting them to get the job done as well as the best architectures, requirements, and designs emerging from self-organizing teams. The dimension of uncertainty avoidance reflects the agile value of adapting to change over following a plan, as well as the ideals of welcoming changing needs, even late in development, and working software over thorough documentation.



Clan-type organizations are characterized by flexible structures, facilitating leaders, loyalty, and cohesion, according to Siakas and Siakas (2007). Clan offers great power distance and weak uncertainty avoidance. Strong power distance and high uncertainty avoidance are characteristics of hierarchical organizations. They rely on vertical hierarchy, strong leadership, formality, and status. Power distance and uncertainty avoidance are weak in a democratic culture. Coordinating leadership, flexible rules, and encouraging employees to contribute to decision-making are typical features of democratic cultures. Weak power distance and strong uncertainty avoidance characterize the disciplined culture. This kind of culture emphasizes formal structure and rules, self-discipline, and self-control. The best way to implement these dimensions within agile methods has been assessed in the prior publications. Agile method application is simplest in democratic cultures, according to Hajjdiab et al. (2012), while it can be challenging in high power distance cultures, according to MacGregor et al. (2005). Hirsch (2005) discovers that it is difficult to transition from a plan-driven culture to agile development.

While the evidence reviewed above emphasizes that only certain types of organizational culture improve prototypical agile practice adoption, some studies have claimed that different types of organizational culture can support certain components of agile practices in different ways (e.g., Anjani et al., 2021; Bunyakiati & Surachaikulwattana, 2016; livari & livari, 2011). Although these studies provide new insight into the possibility that various organizational cultures may support agile practices in various ways, little is known about how organizational cultures and agile practices can change one another since many of the studies mentioned above have focused on the archetypal agile practices as the ideal. Some of the earlier assessments are static in that they assume either organizational culture or agile practices as givens and do not investigate how the archetypal agile practices and organizational culture are both transformed – a gap that we strive to fill. Based on the literature review on agile methods and organizational culture above, we try to further this research tradition by studying how agile methods are fitted in different organizational cultural settings. With this aim in mind, this study examines how organizational culture and agile methods interact and are mutually adjusted. We study how the variance of organizational culture, as classified by Siakas and Siakas (2007) taxonomy, influences the adoption and adaptation of agile practices. We also

investigate how practitioners attempt to adapt organizational culture in the face of isomorphic pressures exerted at the company and industry levels (DiMaggio & Powell, 1983).

## Methodology and Analysis

### Methodology

The data for this study was collected through semi-structured interviews. For example, Leewis et al. (2021) and Radhakrishnan et al. (2022) have applied semi-structured interviews as an IS research method. The first author conducted interviews with teams F, G, and H in Finland. Ethics approval was given by Coordinating Ethics Committee of the Helsinki University Hospital, the approval number 31, 26/13/03/00/13. Both authors conducted interviews in companies A, B, C, D, and E in Japan. The interview themes in Finland and Japan are presented as Appendices A and B respectively. All the interviews were conducted in-person, recorded, and transcribed. Table 1 summarizes the backgrounds of the 50 interviewees.

Table 1 – Summary of the Interviewees					
Company / Team	Country	Industry	Leaders / Managers	Developers / Engineers	Product Owners
Company A (embedded)	Japan	Sub-contractor for car industry	1	1	0
Company B (software)	Japan	Travel booking systems	2	2	0
Company C (software)	Japan	Search engine systems	1	2	0
Company D (software)	Japan	IT system vendor	1	2	0
Company E (embedded)	Japan	Electronics	2	3	0
Team F (software)	Finland	Credit, invoice, and payment	1	5 (including 1 Scrum Master)	1
Team G (software)	Finland	Credit, invoice, and payment	1	15 (including 1 Scrum Master / Sub Team Leader)	1
Team H (software)	Finland	Credit, invoice, and payment	1	7	1

The data from teams F, G, and H consist of interviews from three teams of a software development unit in a large company. So, the entire software unit, or all of this company's software development teams, makes up the Finnish sample. The industry sector of debt-related administrative services as opposed to software development may steer the culture of the company towards risk-avoidance. Companies A, B, C, D, and E developed either embedded systems (companies A and E), IT systems (company D), or web-based information systems (companies B and C), respectively. A specialized computer system called an embedded system is one that is integrated as a part of a larger system, typically with hardware and mechanics, for a specific activity or goal (Kaisti et al., 2013). A key distinguishing characteristic of an embedded system is the need for close hardware synchronization. The physical characteristics of the hardware place restrictions on embedded software. Hence, compared to software development, embedded systems development may be more influenced by the organizations around it. We intended to include companies with different sectoral backgrounds to examine their influences. Although it is difficult for qualitative case

studies to achieve statistical generalization as expected in quantitative studies, the diversity of cases with sufficient background information enables readers to examine the analytical generalizability of our results (Yin, 1994). The agile/lean methods adopted by teams F, G, and H were mixed or hybrid (practices from Scrum, Kanban, and so on). A, B, C, D, and E employed hybrid or mixed agile/lean or agile/waterfall methods.

In teams F, G, and H, agile methods had already been in place for a few years, so that the practices were somewhat established. While the first two self-managed teams, F and H, had been successful in establishing and maintaining agile functioning, the third team, G, had encountered difficulties. The management directed the teams to work in a more top-down and controlled manner. The establishment of agile transformation in the company of the teams F, G, and H was made possible by a large new development process and was based on a recommendation by consultant companies. Each team underwent the shift on its own while working with an agile coach. Although teamwork in team F initially lacked clarity, soon the process had been refined and developed. Team H started applying agile principles and learning by doing. In team G, things were initially going well. Everyone was enthusiastic, and the business side was pleased as well. Subsequently, according to the interviewees, agile development was blamed for the problems on the business side, selling the project unrealistically. The team's adoption of agile was at a standstill at the time of the interview, and fresh ideas for enhancing agile were not being discussed.

Teams in companies A, B, C, D, and E often had people with both little and a lot of experience with agile techniques, except for company D, where every member was highly skilled. The senior members were required to train the junior members on the job because they frequently lacked sufficient practical knowledge. In companies A and B, agile practices were incrementally adopted and adapted to the previous waterfall practices. Teams in company B tried to completely transform from waterfall to agile practices. Teams in companies C and D adopted agile practices from the beginning. Sometimes, there was a lack of coordination with other departments of the company, for example, with hardware, where the waterfall model was adopted, especially in company E. Company D had the highest level of maturity for agile methods, followed by firms B and C, then companies A and E ( $D > B = C > A = E$ ).

### **Validity and Reliability**

Although our research data included only interviews, we also collected other types of data and used triangulation as a member-checking technique and to cross-validate our findings. The supplementary data for companies A, B, C, D, and E included company visits. Within the companies, there was a different emphasis between interviews and company tours in which agile methods were shown in practice. The observations were conducted by the same researchers that conducted the interviews, i.e., the first and the second author. Surveys, physiological measurements, workshops, and key informant interviews were conducted for teams F, G, and H. The first author was involved in these data collections. Although we did not use this data in our analysis directly, it supported the conclusions we reached in this article.

We found semi-structured interviews as a suitable methodology to study our research question. Semi-structured interviews ensure consistency (MacGregor et al., 2005). As the first author participated in all the interviews conducted in companies A, B, C, D, and E with the second author, it was possible to understand and relate to this data first-hand. To gain a broad picture, we interviewed both developers and leaders/managers in all companies. While we were able to interview leaders/managers and developers individually in some of the companies A, B, C, D, and E, in other companies we had to interview them together, which may have affected the views expressed in the interviews. In the interviews of the companies A, B, C, D, and E, the authors asked the leaders or managers to answer their questions by clearly referring to their own experience as a representative agile development project. In this sense, the team serves as the primary unit of inquiry in companies' A, B, C, D, and E and teams' F, G, and H interviews.

Therefore, our analysis and discussion are compatible, despite the use of different frames of data. The data from teams F, G, and H does not comprise several organizations, whereas firms A, B, C, D, and E lack in-depth analyses of a single organization. Convenience samples were used for both data sets. We attempted to diversify the companies in Japan because it was not possible to collect data from multiple teams within one company there because they would not allow it, like we did in Finland. We adopted three teams with diverse characteristics in Finland; therefore, we believe it is possible to mitigate the fact that teams F, G, and H are from the same organization.

The themes of interest in this research were successfully extracted from the data using content analysis, which proved to be a suitable technique. Content analysis itself is a method which enforces the authors to be coherent across the companies analyzed. The combination of theory-informed and data-driven approach in the analysis also worked well, allowing us also to deal with potentially unexcepted findings in a flexible way. The interviews were done by the same researchers – the first and second authors – who also served as raters. The interviews done in Finland were rated by the first author, and the interviews conducted in Japan were rated by the second author. In the content analysis, we attempted to verify that we consistently applied the framework of themes to each case by repeatedly coding each example until the system of coding was "saturated". We discussed how codes are applied to data and agreed how to code the same content. Themes were discussed and compared until a consensus on how to interpret the findings was obtained.

Accordingly, we tried to ensure the validity and reliability in the following ways. To increase the *internal validity* (Yin, 1994), it was first necessary for both authors to have critical discussions about their analyses and interpretations of every case. Also, some of the interviewees were contacted thereafter to confirm the validity of our analyses and interpretation. Secondly, we tried to ensure the diversity of cases providing background information. This way readers can assess the generalizability of our conclusions and interpretation, which enhances the *external validity* (Yin, 1994). Thirdly, we had critical discussions to ensure that we applied the conceptual framework to all the cases in coherent ways, which enhanced the *reliability* of the results and discussion.

## Analysis

We used qualitative content analysis as a key analytical method in our investigation. Content analysis is "a research method for the subjective interpretation of the content of text data through the systematic classification process of coding and identifying themes or patterns" (Hsieh & Shannon, 2005, p. 1278). The field of IS study has made extensive use of content analysis, as Indulska et al. (2012) show. Content analysis as a methodology for analyzing text data in technical domain, according to Anand et al. (2017). Conceptual frameworks serve as a theoretical lens that directs the gathering and analysis of data (Senapathi & Srinivasan, 2012). First, we used the interview themes (Appendices A and B) as a basis for coding. The data were analyzed by coding each theme studied, while the authors were also open to creating codes not expected beforehand, which enabled the generation of novel themes. In other words, we utilized both "conventional" and "directed" content analysis (Hsieh & Shannon, 2005), or both inductive and deductive approach (see also, Finfgeld-Connett, 2014; White & March, 2006).

The NVivo and MAXQDA programs were utilized to help with the coding. We conducted a content analysis using the software, which enabled us to qualitatively code the data. Simmons et al. (2011) have also used computer-based content analysis for qualitative research in the field of information systems. Companies' A, B, C, D, and E data were analyzed by the second author, whereas teams' F, G, and H's data were analyzed by the first. Additionally, the first author was present during the data collection in A, B, C, D, and E and had notes from those interviews. The second author attempted to identify common and divergent characteristics



among companies by coding the transcribed records to conduct a comparative analysis with teams F, G, and H. This was done because companies A, B, C, D, and E consist of teams from several companies with different contexts. From this process, it was possible to obtain the conceptual framework with a firm foundation.

In the next phase, the authors presented their results to each other and discussed them, asking for further clarifications about each other's results. To draw conclusions from the various datasets and to detect common patterns, a comparison was done together. We identified some common and contrasting themes that were worth further exploration. During the analysis, the differences allowed useful contrasts to be made (see also, Senapathi & Srinivasan, 2012). The sections that dealt with the themes of interest were examined further in detail. We then re-inspected our respective data sets and elaborated the findings related to the themes we had selected. We shortlisted and re-organized the themes we found most fruitful in highlighting the contrasts and the similarities between the companies. At this stage, we identified the following themes: individualism and collectivism, generalism and specialism, planning and documentation, division of leadership functions, and continuous improvement. This part of the analysis was both theory-informed and data-driven.

The themes we chose as a framework for our final analysis are shown in Table 2. At this point we focused the analysis on organization culture. The themes of power distance and uncertainty avoidance were chosen as dimensions of our theoretical framework as they were theoretically derived from Siakas and Siakas (2007) categorization. Power distance is examined from two related perspectives, namely, discretion given to the team members and team members' reliance on the leaders. These two sub themes were chosen because they reflect the agile principles of building the project around individuals and trusting them to get the work done and best architecture, requirements and designs emerging from self-organizing teams. The themes of self-organizing teams and leadership are of particular importance in terms of organizational culture fostering agility. The Siakas and Siakas (2007) categorization's uncertainty avoidance factor served as inspiration for the planning and documenting theme. The two sub themes of planning and documentation were chosen because they reflect the agile principles of working software over comprehensive documentation and responding to change over following a plan as well as the agile value of welcoming changing requirements, even late in development. There are illustrative quotations from the themes in Appendix C.

<b>Table 2 – Themes for the Analysis</b>	
<b>Themes</b>	<b>Contents of the theme</b>
Power distance: Discretion	Self-initiative, self-organizing teams and teamwork
Power distance: Reliance on the leaders	Leadership functions (Team leader, Agile Coach, Scrum Master), the degree of leader dependence
Uncertainty avoidance	Heavy or light planning and documentation

Lastly, using Siakas and Siakas' (2007) typology, we classified the organizational cultures of the teams and companies into clan, hierarchical, discipline, and democratic organizational cultures.

## Results

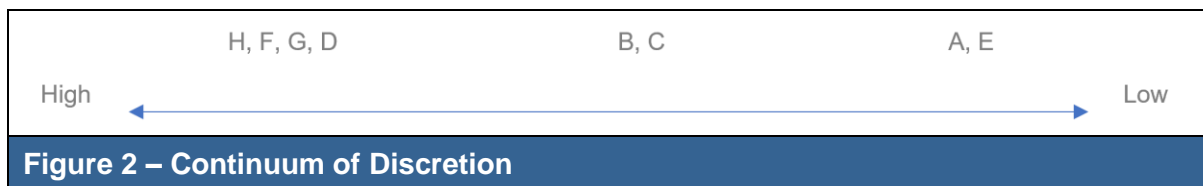
This section presents our findings according to the themes we chose to study.

### *Power Distance – Discretion*

We look at self-initiative, discretion, and self-organizing teams in this part. Figure 2 shows the positions of the companies and teams along a continuum of discretion. The team's scope of

work is expanded to varying degrees in each of the companies A, B, C, D and E. For instance, the team members in companies B and C decide on the requirements jointly to help developers improve their skills. Whereas developers in companies A and E substantially rely on their leaders to make specification changes because they are largely considered as "trainees," who require their leaders' backing, developers at company D are treated as independent professionals who completely address the requirements by themselves. The agile principle of best requirements coming from self-organizing teams is reflected in the team's potential and capacity to oversee requirements. Companies A and E provide developers with the least discretion, whereas company D gives them the most. Companies B and C are situated in the middle and have tried to provide more discretion. Although company D focused on the proactivity of each engineer, companies A, B, C, and E emphasized the collective aspect of agile. Due to the absence of engineers' abilities and initiative, companies A, B, C, and E are only partially able to realize the greater discretion at the individual level brought about by the adoption of agile methods. Discretion in company D and teams F, G, and H, which are from the same organization, supports teamwork, which is also a collective effort. Teams F, G, and H are relatively self-organizing. Individual initiative is also necessary for self-organizing teams. It has been discovered that individual autonomy and team autonomy are connected (Jönsson & Jeppesen, 2013). As the following quotation from team F indicates, self-initiative supports teamwork:

And about the team I can say it is pretty much self-organizing but sometimes needs going back to the basic values and going through the vision that where are we going and so on so that is probably my biggest job but otherwise the team handles everything pretty nicely.  
(Team F, Team Leader)



We define teamwork as “dynamic” when individuals attempt to change how they work together; otherwise, we define teamwork as “static”. Agile methods require dynamic as well as static teamwork. Static teamwork, such as sharing duties and difficulties, works effectively in companies A, B, C, and E. Engineers can consult their peers rather than their superiors by discussing their current problems. In some companies, engineers determine their own tasks using a task board. In other words, it is the team’s responsibility to complete the tasks. However, in contrast to company D, where both dynamic and static teamwork work very well, companies A and E find it extremely difficult to engage in dynamic teamwork, such as suggesting changes and deciding what to do during the sprint the following week. Although they wish to encourage discretion at the team level, companies A, B, C, and E place a great deal of responsibility on the shoulders of the leaders. While dynamic teamwork has steadily grown in companies B and C as they have attempted to increase the developers' proactiveness by altering the role of their leaders, as will be discussed later, their level of dynamic teamwork is not as high as that of company D. Both dynamic and static teamwork work, at least in part, in teams F, G, and H. The most mature team H, however, exhibits the most dynamic teamwork. Teamwork is relatively dynamic in team F, which is partially distributed. Teamwork in team G is less dynamic.

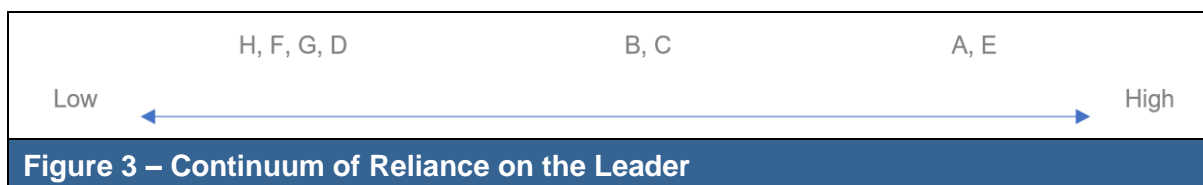
Teams F, G, and H share challenges and successes, and the shared responsibility lessens the pressure on each team member. In agile methods everyone can observe the progress of other team members, as the following quotation from company C illustrates: “In agile practices, everyone can see the progress of other team members. This situation in a sense urges individuals to catch up with the progress of other developers.” (Company C, Developer)

Although team members share the responsibility of the tasks, this causes stress in company C. When agile was first introduced, this was also true for team G. Yet, the change was seen very positively, once they understood they could manage their own workload. In fact, it has been acknowledged that one of the possible drawbacks of agile approaches is constant surveillance and the feeling of being monitored (Annosi et al., 2016). This may be connected to an earlier phase of adopting agile. Personnel appraisals, or the performance rating of each employee that influences pay and promotions, are a reoccurring theme in companies A, B, C, D, and E, but they are largely absent from the conversation in teams F, G, and H. Multi-skilling is emphasized in teams F, G, and H and is thought to lessen individual pressure and workload. Companies A, B, C, D, and E work to create engineers with multiple skills to enable task sharing in a team. Because of the organizational structure, it is challenging to establish a cross-functional team in company A. Self-organizing teams have a hurdle due to the absence of cross-functionality (Hoda & Murugesan, 2016).

Companies A, B, C, D, and E aim to create teams that are generally fixed in their membership. Team building efforts are prioritized. While they acknowledge that some team members will inevitably leave on short notice, they partially succeeded in establishing fixed team members. In some companies, pair programming is an example of “making engineers redundant”. Engineer redundancies are unavoidable because HRM sections prefer job rotation while practitioners value a team's stability. A team can only exist for a maximum of six months in one company. Employee multi-skilling and evaluating candidates for promotion are causes for the rotations. Team H is stable, while the composition of teams F and G has changed, which has allowed for the spreading of competence across teams. The teams seek to stay away from relying solely on one person's expertise. By using the same methods for the outsourced teams, continuity has also been an objective for those teams.

### **Power Distance – Reliance on the Leader**

Figure 3 shows how the companies and teams are positioned along a continuum of reliance on the leader. According to Lazwanthi et al. (2016), power and bureaucratic cultures prevent the adoption of the agile methodology, which calls for a reduction in the reliance on leaders and solving as many problems as possible by horizontal interactions. Teams F, G, and H have the following roles defined: The agile team's official supervisor is the team leader. The Scrum Master, who is typically a team member rather than the team leader, assists the team in adhering to agile principles and practices. The Scrum Master role may occasionally be realized by other roles in teams F, G, and H, such as technical lead and client contact person. The Agile Coach is a term used to describe an internal employee, such as the Scrum Master, in companies A, B, C, D, and E. The Agile Coach and a development leader share leadership responsibility in some of the companies A, B, C, D, and E. The team's official supervisor is the development leader. There are separate Agile Coaches and development leaders in companies B and C. Agile Coaches are either nonexistent or scarce in companies A and D. The positions of the Agile Coach and the developer leader are integrated in Company E. It has been noted that roles that overlap, such as Scrum Masters serving as project managers, are problematic (Jovanović et al., 2017).



**Figure 3 – Continuum of Reliance on the Leader**

The presence of a Scrum Master varies between teams in teams F, G, and H. The role of the Scrum Master causes confusion in team G because the organization does not support it and because Scrum Masters only periodically exist. Team F has a designated Scrum Master, and

some tasks typically done by a Scrum Master, such as removing obstacles are carried out by the team members themselves. Mature team H decided against having a Scrum Master because they believe all members can carry out these duties and they want to promote joint responsibility, for example, in meetings with the Product Owner and when planning and moderating meetings. They believe not having a Scrum Master is related to the team's capacity for self-organization, its courage to take responsibility, communication and trust within the team – including the freedom to discuss problematic issues as well. Instead of joint leadership, they refer to it as joint responsibility. In fact, it has been noted as a concern that a Scrum Master limits self-organizing (Jovanović et al., 2017). Developers feel that having a Scrum Master might be good but the things missing are mostly seen as result of the team's "laziness", rather than due to the lack of a Scrum Master. On whether there is a need for a Scrum Master, the team's opinions were divided, however.

The Scrum Master's role has occasionally been combined with a team leader's role in teams F, G, and H, but the team leader's authority does not align with the Scrum Master role, as the following quotation from team G suggests:

...Scrum Master is the one to whom the developers should be able to come to talk freely if something is not alright and so on. So then if s/he is also your supervisor, for some people it can set some restrictions and result in a particular kind of attitude, so that it's not a good combination that the Scrum Master is also a supervisor. (Team G, Developer)

Nonetheless, as the following quotation implies, some interviewees felt that the duties of a Scrum Master did fit those of a supervisor in teams F, G, and H:

Interviewer: How do you think combining being a Scrum Master and a supervisor works?  
Interviewee: I feel for my part it has worked. I have been a Scrum Master for these same people for so long and then when I was a Scrum Master for a larger group so that I think it is at least for my part the job description was very similar, taking care of and such... leadership in a way, even though... you don't have that kind of command on anything... (Team G, Scrum Master/Sub Team Leader)

For instance, team leaders can utilize their positions of authority to promote agile practices and protect their team inside the company. The quotations serve as an example of how the sub-team leader/Scrum Master had a different perspective from a Team G developer. The developer was more critical of the mix of a supervisor and Scrum Master roles than the Scrum Master/sub team leader.

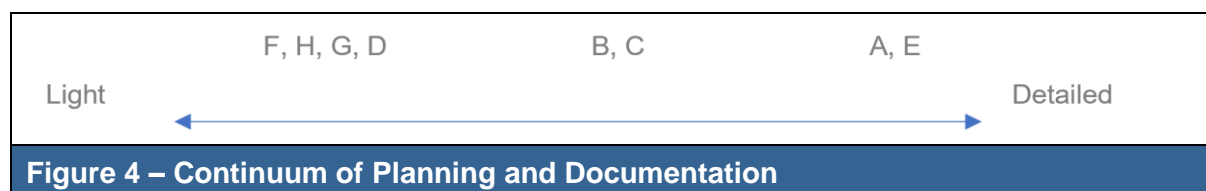
The interviewees believe that team F's supervisor participating in the technical solutions is not necessarily beneficial for the self-organizing team because according to agile principles the team is supposed to fulfil the demands from the business. Also, team leaders in companies B, C, and E aim to avoid involvement in development work. For example, in company C, team leaders abstain from participation in development work to avoid reducing the proactiveness of members. The company also implemented a policy to switch from "teaching" to "coaching" as a leadership style to encourage members to take the initiative. The role of leaders reflects the notion of agility about building projects around motivated individuals and trusting them to get the job done. A challenge for self-organizing agile teams' project management has been recognized as asserting autonomy, for instance, while sharing project management responsibilities like taking part in eliciting requirements (Hoda & Murugesan, 2016). Under the leader's absence, it has been nearly impossible for companies B and C to move forward with development. But things are evolving. Problem-solving reliance on the leader has decreased, while interaction among engineers has grown.

The team leader in companies A, B, C, D, and E must be skilled at both the technical competences and motivating people. While leaders need to be self-driven, proactive

engineers, they are frequently in short supply. Team leaders in these companies have too many responsibilities, particularly in companies A and E with little prior experience with agile. Their challenge in teams F, G, and H is not having enough time with the team. Team leaders have more responsibilities in companies A, B, C, and E than in teams F, G, and H. As agile methods require motivated individuals to collaborate (Agile Manifesto, n.d., <https://agilemanifesto.org>), whether and how leaders are involved in HRM-related matters is an important question. Team leaders in companies A, B, C, D, and E oversee HRM-related matters like capability development and employee evaluation. Although the team leaders are officially in charge of competence development in teams F, G, and H, in reality the team members frequently take on this task themselves, in contrast, in companies capability development.

### **Uncertainty Avoidance – Planning and Documentation**

Agile methods do not place a strong emphasis on thorough planning and documentation because changes in the requirements are inevitable (Hajjdiab & Taleb, 2011). Figure 4 shows the positions of the companies and teams along a continuum of planning and documentation. Companies A, B, C, and E place a lot of emphasis on planning capacity. One reason for this is that they give a lot of importance to fine-grained analysis and planning. Another is that the level of skills may not be high enough for agile practices because agile requires planning capacity. They claim that once agile development is underway, it is simpler to avoid doing too much work because the workload is examined and negotiated in detail. This allows as precise a plan for a sprint as possible. According to company E, each sprint involves a "little waterfall," therefore the waterfall experience is still significant, as the following quotation suggests: "One cannot practice agile without previous experience in waterfall development because the agile development process can be seen as a series of small waterfalls. In each small waterfall in the agile process, one should know how to practice waterfall development." (Company E, Leader). Although they attempted, company A found it difficult to totally avoid specialization. They were trying to acquire competencies, especially estimation and planning skills.



**Figure 4 – Continuum of Planning and Documentation**

Moreover, company E has developed a practice of thorough documenting that was not present during the waterfall. Following the implementation of agile, detailed planning is stressed more than it was during the waterfall era in company C:

When we adopted the waterfall model, we tried to develop all functions in a certain period, which often left some functions undeveloped. We started to divide functions into much smaller units after adopting agile development to avoid this kind of 'leakage'. Company C, Agile Coach)

This citation demonstrates the uncertainty that practitioners aim to avoid by carefully preparing ahead. It is interesting that a higher level of planning and documentation is seen in agile than waterfall, contrary to the Agile Manifesto (n.d.) (<https://agilemanifesto.org>) which emphasizes working software and responding to change over comprehensive documentation and following a plan.

There are times when it seems like the documentation for teams F, G, and H is a little lacking. There is a perception that planning is often not fine-grained enough, yet they wish to avoid unnecessarily detailed planning. The most planning and documentation were stressed by

team G since they believed the surrounding organization was blaming agile methods for issues. Daily meetings occasionally converted into planning and problem-solving sessions in team H, the agile team that was otherwise the most mature. Planning sessions had previously been overly lengthy and comprehensive. Prolonged estimation efforts and accurate estimations are a challenge (Hoda & Murugesan, 2016), and lengthy meetings have been listed as barriers for an agile team's decision making (Drury et al., 2012). Teams F, G, and H view product vision and long-term release planning, or visibility beyond one sprint, as troublesome. According to Jovanović et al. (2017) problems can arise from the absence of a shared product vision.

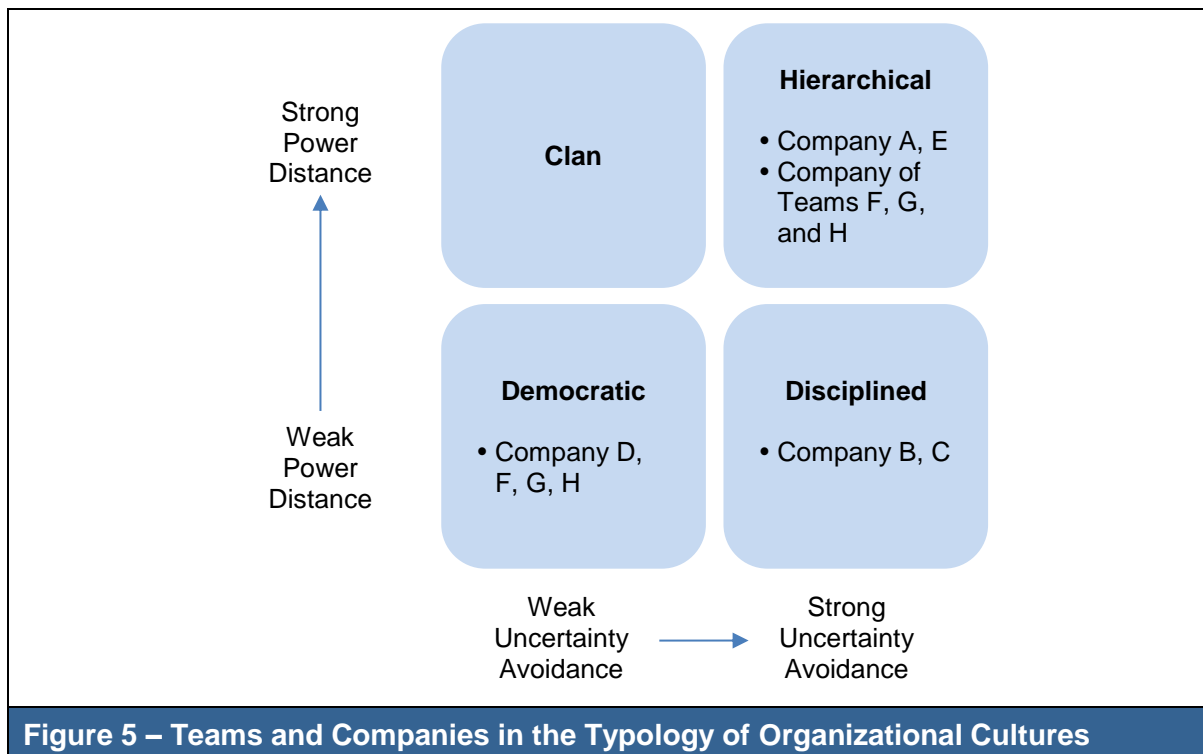
Table 3 lists the analysis's findings in brief.

Table 3 – Results of the Analysis	
Themes	Variety between companies
Power Distance – Discretion	<p>Self-organizing collaboration necessitates individual initiative, which is seen as a severe limitation: Companies A and E are most affected by a lack of skills or proactiveness, whereas companies B and C are modestly affected. Functioning self-organizing teams, team spirit, and communication (teams F and H), and supporting environment (company D); partly functioning in team G.</p> <p>The difficulties in dynamic cooperation are felt most clearly in companies A and E, and less so in companies B and C. Static teamwork, such as helping or sharing tasks and task board, functioning. Not a special problem in company D. Individual pressure is reduced through shared responsibility, control over the amount of work, and multi-skilling (teams F, G, and H).</p> <p>Teams may not always be stable because of worker dispatch, rotation, and the introduction of redundancy in companies B, C, and E. Several team compositions increase team competence in teams F and G. The ideal team has stability in team H.</p>
Power Distance – Reliance on the leaders	<p>Concentration of tasks with leaders, the position of the leader is important in companies A and E. Limited role of leaders in company D. Change in leadership style supporting proactiveness, from teaching to coaching in companies B and C. Among teams F, H, and G, team H had the lowest reliance on the leader, also low in team F while team G had a somewhat greater level.</p> <p>The division of leadership is unclear in companies A and E. Supervisor role sometimes like that of Scrum Master in teams F, G and H. The presence of a Scrum Master varies in teams G and H.</p>
Uncertainty Avoidance – Planning and documentation	<p>Companies A and E placed a great emphasis on fine-grained planning, whereas companies B and C placed less emphasis on meticulous planning and company D did not emphasize it. Engineers with strong design and planning skills are needed in companies A, B and E. Companies C and D have fewer or no issues with that. Avoiding intensive planning in teams F, G, and H; occasionally, daily and planning meetings are overly elaborate and drawn-out in teams F and H. Lack of long-term release strategy and product vision in teams F, G, and H.</p> <p>Comprehensive documentation in companies A and E. Less detailed documentation (Companies B, C). Light documentation in company D, teams F, G, and H, with team G placing the most emphasis on documentation. Teams can refuse the assigned deadline since they began to carefully plan the sprint in companies B, C, and E.</p>

### Organizational Culture

Finally, we describe the organizational culture of the teams and companies we have studied. For the purpose of demonstrating the distinctive organizational cultures of each company, we

employ Siakas and Siakas' (2007) typology. We assume that power distance (PD) is high when discretion is low and leaders play a significant role, and that uncertainty avoidance (UA) is high when an organization significantly relies on planning and documentation. Figure 5 shows the positions of the teams and companies within the typology of organizational culture.



First, we examine the organizational culture surrounding the teams. The organizational cultures of companies A, B, C, D, and E are as follows: companies A and E are hierarchical, companies B and C are disciplined, and company D is democratic. Company A is a risk-averse organization because it is situated in a supply chain of subcontractors that is highly managed and stresses "zero-defect" in the automotive industry. Company E has a long history of about 100 years. Electronics and household appliances make up its core line of business. Because software is a relatively new field, software developers are relatively free and separated from the company culture dominated and driven by hardware developers. They started implementing agile methods in part because of this. Companies B, C, and D were all founded in the 1990s as ICT businesses; as a result, their corporate cultures differ somewhat from that of bureaucratic businesses. B and C have a more bureaucratic feel than D because they developed quickly into large companies.

Companies A and E have hierarchical cultures with a high level of power distance and uncertainty avoidance. Because of this, many of the developers in this area have limited knowledge and expertise, and their managers also oversee on-the-job training in addition to project management. Thus, the power distance in these companies is high. Due to their lack of skills, developers rely on the assistance of leaders. Their emphasis on detailed plans means that their uncertainty avoidance is also high. Their hierarchical culture is also related to both developing embedded systems, because their users – manufacturers of automobiles for A and consumer electronics product for E – require quite high-quality software. This relates to technical challenges in implementing agile approaches for embedded software, since engineers find it challenging to acquire many fields of expertise because it necessitates in-depth, specialized knowledge. The implementation of the self-organizing principle is then difficult for them (leading in a high PD) due to a technical reason and planning and the division of labor tend to be meticulous (high UA).

Companies B and C have similar organizational cultures that are disciplined. Their reliance on leaders as supervisors is less than in companies A and E, which suggests that the power distance of these companies is lower than in A and E. Simultaneously, both attempt to shift from a supervisory to a facilitative leadership style, which can be interpreted as an effort to reduce the power distance further. Like companies A and E, they stress detailed planning and see agile processes as a sequence of little waterfalls. As a result, their uncertainty avoidance is as high as companies A and E. They do seek to embrace a democratic culture by lowering uncertainty avoidance by, for example, reducing reliance on detailed planning.

The organizational culture of company D is democratic. Power distance is low because this organization relies so significantly on highly talented and independent developers. Company D has multiple-skilled developers because mid-career hiring is the primary method to obtain developers, as evident in the following quotation:

In fact, it usually takes at least three years until a non-experienced developer becomes a full-fledged developer who knows the latest technology, programming languages, and development methods. So, we gave up such training efforts and hired only experienced developers. Therefore, we almost totally depend on talented individual developers when we must solve problems. In other words, we do not try to solve problems in systematically organized ways. (Company D, Leader)

Therefore, it does not depend on detailed planning, indicating that uncertainty avoidance is also low.

Teams F, G, and H come from the same company. Their company's culture can be described as hierarchical with high power distance and uncertainty avoidance. The company outside of software development has often operated in a waterfall process with distinct functions. It has been difficult for this organization to grasp and value agile and the agile attitude. Support for agile has varied; in the past, management even attributed difficulties to agile; other times, however, management has been oriented toward agility. The developers see that there is occasionally a split between IT and business and that software development and the rest of the company have separate value bases. The objectives of the developer teams are understood differently than they are by those in charge of business and operations. The customer interface has non-agile actors, and feedback loops have been a challenge. The business side is not used to reacting quickly and it is difficult to obtain answers from them swiftly. The sales department has sold projects using the waterfall method without understanding of agile, which is evident in the following quotation:

Traditionally it has been like that the sales have asked customer "what do you want?" and then written up all that the customer wants and then gone to the development team and told them "This has been sold and it has to be ready by this time". And if the timetables are not met it is of course the development team's fault [laughter]. It is a bit bad equation as it seldom matches. (Team H, Developer)

Sometimes agile roles, such as Product Owner and Scrum Master, have been problematic and caused confusion, as the organizational context behind them is unsupportive. For instance, the backlog decision-making authority of Product Owners has changed over time. Developer transparency does not align with the organization's overall culture. The developers believe this because their openness goes against the culture of the rest of the company and occasionally works against them.

The sub-organizational cultures of teams F, G, and H are distinctive from one another and the organizational culture as a whole. Even in the field of software development, there are different perspectives on agile; the three development teams do not share a common approach to achieving it. The team culture of team F is an example of a democratic culture with low power



distance and uncertainty avoidance. The immediate supervisor takes a coaching approach towards the team members. The distributed team members, with whom they only infrequently collaborate, and the fact that not all of the distributed teams are agile provide difficulties for this team. They have, nevertheless, been successful in developing some functioning practices for distributed teamwork. The team is prepared to change directions quickly. Sprint goals and timetables create predictability. When it comes to the strengths and the challenges of the team, for example, possessing clear vision both as a team and as an organization, and distributed teamwork, the team leader and developers share similar perspectives.

Team H's team culture can also be characterized as democratic. Teamwork is agile, as illustrated by the following quotation:

On the other hand, I would say that people have a pretty good perception of what agility is and a will to do things right and well. So, to some extent agile values are in a way close to our hearts... learn new things and always do better. So you could at least summarize it. (Team H, Team Leader)

With pair programming and learning from other team members, competency development is achievable. All the team members have similar roles and sufficient skill to share responsibilities. The team places a strong emphasis on openness and transparency, and it is safe to test out new ideas. The supervisor supports and coaches the team members while organizing and facilitating continuous improvement. Instead of getting involved in the technical solutions, he lets the team handle the demands of the business in line with the agile principles.

The culture of Team G is likewise one of democratic. The team can respond and alter course rapidly, and sprints and coding periods facilitate efficient work. This is in line with the agile philosophy of adapting to change rather than following a plan. A backlog allows for transparency, and agile methodologies have improved communication. The members of the team share in the tasks and duties, which improves team spirit. However, because it conflicts with the organization, this team's condition is the most challenging. Because of a long and complicated customer project, the management imposed stricter control, and the team faced layoffs, as the following quotation illustrates:

At some point, I don't know why, the sprints were removed... And I didn't like it when I came here, I didn't like it at all because I think it threw away a little bit the agile idea that you get something done continuously and there are short milestones... So, I wanted them back... Now that we have these layoffs, we don't have sprints either... (Team G, Team Leader)

Stress and a lack of motivation have resulted from layoffs. Estimates from the developers are not always used, and the product backlog is occasionally not discussed with the developers. It is not always possible to influence working time. For the outsourced team members, self-initiative is a challenge. Scrum Masters have been requested by developers to promote agile. Challenges include testing new solutions and engaging in continuous improvement. In team G, there are both similarities and differences between the experiences of the team leader, the sub team leader/Scrum Master, and developers. For example, the team leader realized the difficult situation the team was in, in a similar manner. Yet, compared to the sub team leader/Scrum Master and developers, the team leader was slightly more critical of agile methodologies. It may have something to do with the team leader being in a tough position between the team and management, who saw agile methods as partially to blame for the team's challenging circumstances.

In summary: 1) Adopting agile methods does not necessarily cause convergence of organizational and team culture toward democratic which seems to be most compatible with archetypical agile practices. Companies and teams can also adapt agile practices to

organizational and team cultures which are, in some cases, difficult to change, as in the cases of companies A and E. 2) At the same time, organizational and team cultures can also be adapted toward democratic, as shown by the examples of the companies B, C, and D as well as the teams F, G, and H.

## Discussion

When studying the relationship between agile methods and organizational culture, some prior research has assumed either agile practices or organizational culture as given, but little is understood about how they both are mutually transformed. We show that companies A and E initially attempted to adapt agile practices to their hierarchical organizational culture, indicating that organizational culture may occasionally be seen as an exogenous variable determined by elements such as firm-, product-, and industry-specific features (cf. DiMaggio & Powell, 1983), for instance. We might also argue that these situations represent the evolution of unique agile techniques as opposed to immature ones. In fact, like in the instance of company E, the fact that they struggled with a major lack of planning ability while enthusiastically pushing agile methods demonstrates that their agile practices diverged from the archetypal agile as anticipated in the Agile Manifesto (n.d.) (<https://agilemanifesto.org>). Actually, one of the major difficulties in implementing agile in these businesses appears to be the apparent lack of planning skills, as an example of a high-level expert competence. In prior literature, one of the problems with implementing agile was in fact a lack of skills (Faisal Abrar et al., 2020). The issue, however, is not necessarily the engineers' objective skill level, which is unknown given this data, but also whether agency is given to and/or taken by the engineers. Uncertainty-avoiding organizational culture, which places an emphasis on planning capacity because the organization seeks to avoid uncertainty by planning more thoroughly, may also be a contributing factor to the perception of a lack of planning abilities. In addition, we could contend that the scarcity of experienced engineers is supplemented by a significant reliance on planning capacity.

Second, we demonstrate that teams and companies can still attempt to change the culture to one that is democratic, which appears to be most compatible with the archetypal agile principles, even when they are surrounded by a culture that does not align with those practices. Some of the prior studies demonstrate that various organizational cultures may reinforce different aspects of agile practices (e.g., livari & livari, 2011). Although their argument is important in that they go beyond the assumption of a one-to-one relationship between agile method and democratic organizational culture, they do not assume variability of organizational culture. The cases of companies B, C, and D show that organizations and teams can change their culture in part through human resource management strategies, such as in-house training, rotation practices, leadership style adjustments, and an unusually high reliance on mid-career hiring in the context of a labor market where long-term employment predominates. They were able to avoid isomorphic pressures brought on by the surrounding labor market, for example, by taking these actions. It should be recognized, nevertheless, that their attempts to adapt to the culture inevitably have some limitations. For example, mid-career hiring, as used by company D, may be limited if other companies desire to keep their highly experienced engineers. On the other hand, in-house training has its limitations when there are time constraints, like speedy delivery. Indeed, one of the potential obstacles to adopting agile has been identified as being human resource issues (Wang et al., 2012). Due to the early stages of agile development, cultural change may also be challenging. In immature agile teams where self-organization is not yet viable, reliance on the leader is typical. Later on, the leader's responsibilities become more consultative, facilitative, and coordinative. To promote empowerment and autonomy, it is less directive, command, and control in nature (Gren et al., 2017; Senapathi & Srinivasan, 2012).

In addition, we see that the team microculture at the base of an organization is not entirely determined by the macro culture at the top. The data shows the existence of organizational subcultures, such as divisions between departments or divisions between teams and the organization (for example, embedded vs. software development or software team vs. the rest of the organization). Even when employees are fully integrated into a company with a strong organizational culture, they nevertheless tend to have different perspectives in different teams and divisions. This variation may be partially explained by the actions of local managers and supervisors (Edmondson & Lei, 2014). Our analysis suggests that variance of teams in an organization may depend on the leadership. Companies B and C made a concerted effort to change their leadership styles as a company policy, which very certainly led to a decrease in team variance. The company culture that affected teams F, G, and H was hierarchical in nature; nonetheless, each team tended to effectively create democratic subcultures. Two teams managed to maintain a somewhat distinct subculture from the surrounding organization culture, which was even symbolic, as software development resided in another building from the rest of the organization. A lengthy and problematic customer project, however, caused the third team to clash more with the surrounding organization, and their efforts to foster a democratic team subculture were undermined by the influence of the hierarchical organizational culture, which represented the financial sector's professional culture rather than software development.

This article did not address the national and professional culture, as the sample did not allow us to make conclusions about them, but some tentative issues can be raised for further studies. For instance, Merhi (2021) discovered that IT adoption is impacted by power distance and uncertainty dimensions of a national culture. Li et al. (2022) notes in their article that cultures in the Asia-Pacific area are typically high-power distance, respecting authorities. Siakas and Siakas (2007) propose typical countries for each organizational culture, such as hierarchical for Japan. In contrast to being one of the most uncertainty-avoidant nations in the world, Japan is borderline hierarchical, and the power distance is not exceptionally great, according to Hofstede's cross-cultural comparison (Hofstede Insights, n.d.). Nonetheless, we might contest the premise that national culture determines organizational practices. As for professional culture, different labor market systems tend to generate differences in engineers' capabilities. Japanese labor market institutions place a strong emphasis on long-term employment with internal training and the hiring of recent graduates. Engineers have been viewed as generalists who should manage both engineering and management issues (McCormick, 1996). Companies can find more capable, proactive engineers by upping the percentage of mid-career hiring or by putting in place an internal career system that enables more engineers to advance on the specialist path. This strategy is very similar to what Indian ICT companies systemically pursued, which was to develop elaborate human resource management practices while emphasizing mid-career hiring (Hirakawa et al., 2013). Verifying these sorts of outcomes, however, are outside the scope of this study and require more research, despite the fact that they have substantial consequences for comprehending agile practices in the Asia-Pacific cultural backdrop.

As a limitation, we cannot generalize the results beyond these companies to other software or embedded companies. More research is necessary to validate these results in other companies. Another limitation in our analysis may be that in addition to organizational cultural differences, the differences between the companies are due to the agile immaturity of some companies. Even though some of the organizations have been using agile for a longer time, this may show the necessity for more thorough long-term observation. On cross-cultural issues in global software development, there is prior literature (e.g., Lazwanthi et al., 2016). Few studies, however, compare the use of agile methodologies in various country contexts. There is a void in the literature about the influence of national culture, and more research is required. In addition, there are not many studies that look at agile methods and professional cultures, and the definition of professional culture is still hazy, which also calls for further exploration.

## Conclusion

In conclusion, contrary to what some earlier research claimed, there is not a single ideal combination of agile methods and organizational culture. There are aspects of each culture that support or hinder the adoption of agile methods. This finding is consistent with some of the earlier literature (e.g., Anjani et al., 2021; Bunyakiati & Surachaikulwattana, 2016; livari & livari, 2011). In uncertainty-avoiding organizational culture, long-term orientation encourages planning and documentation, but hierarchical and plan-driven organizational culture can also impede dynamic teamwork that depends on individual capability. Plan-driven approaches are more long-term and agile development more short-term in welcoming changes (MacGregor et al., 2005). In a culture that is low on uncertainty avoidance, short-term orientation may support welcoming changes in an agile way, but it may hamper planning and documentation. In low power distance cultures, discretion encourages individual initiative as the cornerstone of teamwork. In an agile team, both teamwork and individual initiative are required (Gren et al., 2017).

High discretion, self-initiative and self-organizing teams that enable dynamic teamwork, and low power distance, which results in less focus on the leader, appear to be characteristics of democratic cultures that favor agile methods. In terms of agile methods and dynamic teamwork, hierarchical cultures appeared to be the most problematic: large power distance means great reliance on the leaders and low discretion, and high uncertainty means extensive planning and documentation. Agile methods with dynamic teamwork may be better fitted to a non-hierarchical setting in an environment where each employee may voice their opinions. Democratic culture seemed thus favor applying methods while hierarchical culture appeared to somewhat impede it, which is consistent with earlier studies (e.g., Hajjdiab et al., 2012; Hirsch, 2005; Lazwanthi et al., 2016; MacGregor et al., 2005). Because of their lower reliance on the leaders, increased discretion, and reduced power distance, disciplined cultures were partially suited to agile approaches. On the other hand, they emphasized heavy planning and documentation, a feature of high uncertainty avoidance not so suited with agile methods. Clan type of culture was not identified in our data.

The contribution of this study was to identify some emerging themes that may be of relevance when analyzing how agile approaches are used in these various corporate culture contexts. We also add to the Siakas and Siakas (2007) dimensions by providing a more in-depth qualitative analysis that is framed around the underlying issues of the various dimensions. By demonstrating that organizational culture is not uniform and that various subcultures exist instead, we have attempted to address criticism of a part of the organizational culture literature (such as Schein, 2010). We have also shown that culture is not static but dynamic. For instance, there are intentional attempts to transform hierarchical organizational cultures into democratic ones, or the endeavor to uphold a democratic subculture is constrained by a hierarchical organizational culture. Because organizational cultural differences are so ingrained, practitioners will inevitably expend significant effort to adapt agile principles to the local context.

## Acknowledgements

This work was supported by NITech Frontier Research Institute [no grant number]; Japan Society for the Promotion of Science KAKENHI under Grant [15K03426]; the Finnish Funding Agency for Technology and Innovation under Grant [40191/12]; the Association for Promotion of Occupational Health [no grant number]; and the Academy of Finland under Grant [316538].

## References

- Anand, O., Srivastava, P. R., & Rakshit, A. (2017). Assessment, implication, and analysis of online consumer reviews: A literature review. *Pacific Asia Journal of the Association for Information Systems*, 9(2), 43-74.
- Anjani, R. F., Raharjo, T., Hardian, B., & Suhanto, A. (2021). Measuring the relationship between organizational culture and project success: A survey of agile software development teams. *Proceedings of the International Conference on Advanced Computer Science and Information Systems 2021*. Depok, Indonesia: IEEE.
- Annosi, M. C., Magnusson, M., Martini, A., & Appio, F. P. (2016). Social conduct, learning and innovation: An abductive study of the dark side of agile software development. *Creativity and Innovation Management*, 25(4), 515-535.
- Artemeva, N., & Freedman, A. (2001). 'Just the boys playing on computers': An activity theory analysis of differences in the cultures of two engineering firms. *Journal of Business and Technical Communication*, 15(2), 164-194.
- Bunyakiati, P., & Surachaikulwattana, P. (2016). Fit between agile practices and organizational cultures. *Proceedings of the 13th International Joint Conference on Computer Science and Software Engineering 2016*. Khon Kaen, Thailand: IEEE.
- Cram, W. A. (2019). Agile development in practice: Lessons from the trenches. *Information Systems Management*, 36(1), 2-14.
- DiMaggio, P. J., & Powell, W. W. (1983). The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields. *American Sociological Review*, 48(2), 147-160.
- Drury, M., Conboy, K., & Power, K. (2012). Obstacles to decision making in Agile software development teams. *Journal of Systems and Software*, 85(6), 1239-1254.
- Edmondson, A. C., & Lei, Z. (2014). Psychological safety: The history, renaissance, and future of an interpersonal construct. *Annual Review of Organizational Psychology and Organizational Behavior*, 1(1) 23-43.
- Faisal Abrar, M., Sohail, M., Ali, S., Faran Majeed, M., Ali Shah, I., Rashid, N., & Ullah, N. (2020). De-motivators for the adoption of agile methodologies for large-scale software development teams: An SLR from management perspective. *Journal of Software: Evolution and Process*, 32(12), 1-20.
- Finfgeld-Connett, D. (2014). Use of content analysis to conduct knowledge-building and theory-generating qualitative systematic reviews. *Qualitative Research*, 14(3), 341-352.
- Fitzgerald, B., Stol, K. J., O'Sullivan, R., & O'Brien, D. (2013). Scaling agile methods to regulated environments: An industry case study. *Proceedings of the 35th International Conference on Software Engineering 2013*. San Francisco, USA: IEEE.
- Gren, L., Torkar, R., & Feldt, R. (2017). Group development and group maturity when building agile teams: A qualitative and quantitative investigation at eight large companies. *Journal of Systems and Software*, 124, 104-119.
- Gupta, M., George, J. F., & Weidong, X. (2019). Relationship between IT department culture and agile software development practices: An empirical investigation. *International Journal of Information Management*, 44, 13-24.
- Hajjdiab, H., & Taleb, A. S. (2011). Adopting agile software development: Issues and challenges. *International Journal of Managing Value and Supply Chains*, 2(3), 1-10.
- Hajjdiab, H., Taleb, A. S., & Ali, J. (2012). An industrial case study for Scrum adoption. *Journal of Software*, 7(1), 237-242.

- Hirakawa, H., Lal, K., Shinkai, N., & Tokumaru, N. (Eds.). (2013). *Servitization, IT-ization and Innovation Models: Two-stage Industrial Cluster Theory* (1st ed.). London, England: Routledge.
- Hirsch, M. (2005). Moving from a plan driven culture to agile development. *Proceedings of the 27th International Conference on Software Engineering 2005*. St. Louis, USA: IEEE.
- Hoda, R., & Murugesan, L. K. (2016). Multi-level agile project management challenges: A self-organizing team perspective. *Journal of Systems and Software*, 117, 245-257.
- Hofstede, G. (1980). *Culture's Consequences: International Differences in Work Related Values*. Sage Publications: London and Beverly Hills.
- Hofstede, G. (2011). Dimensionalizing cultures: The Hofstede model in context. *Online Readings in Psychology and Culture*, 2(1), 1-26.
- Hofstede Insights. (n.d.). *COUNTRY COMPARISON TOOL*. Hofstede Insights. <https://www.hofstede-insights.com/country-comparison-tool?countries=japan>
- Hogdson, D., & Briand, L. (2013). Controlling the uncontrollable: 'Agile' teams and illusions of autonomy in creative work. *Work, Employment and Society*, 27(2), 308-325.
- Hsieh, H. F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research*, 15(9), 1277-1288.
- Hussman, D. (2004). How to maintain and promote healthy agile culture. In: C. Zannier, H. Erdogmus & L. Lindstrom, L. (Eds.), *Extreme Programming and Agile Methods - XP/Agile Universe 2004, Lecture Notes in Computer Science, vol 3134* (pp. 190-191). Berlin, Heidelberg: Springer.
- Hyun, Y., Kamioka, T., & Hosoya, R. (2020). Improving agility using big data analytics: The role of democratization culture. *Pacific Asia Journal of the Association for Information Systems*, 12(2), 35-63.
- Iivari, J., & Huisman, M. (2007). The relationship between organizational culture and the deployment of systems development methodologies. *MIS Quarterly*, 31(1), 35-58.
- Iivari, J., & Iivari, N. (2011). The relationship between organizational culture and the deployment of agile methods. *Information and Software Technology*, 53(5), 509-520.
- Indulska, M., Hovorka, D. S., & Recker, J. (2012). Quantitative approaches to content analysis: Identifying conceptual drift across publication outlets. *European Journal of Information Systems*, 21(1), 49-69.
- Jönsson, T., & Jeppesen, H. J. (2013). Under the influence of the team? An investigation of the relationships between team autonomy, individual autonomy and social influence within teams. *The International Journal of Human Resource Management*, 24(1), 78-93.
- Jovanović, M., Mas, A., Mesquida, A. L., & Lalic, B. (2017). Transition of organizational roles in agile transformation process: A grounded theory approach. *Journal of Systems and Software*, 133, 174-194.
- Kaisti, M., Rantala, V., Mujunen, T., Hyrynsalmi, S., Könnölä, K., Mäkilä, T., & Lehtonen, T. (2013). Agile methods for embedded systems development: A literature review and a mapping study. *EURASIP Journal on Embedded Systems*, 1-16.
- Kakar, A. K. (2017). Investigating the motivating potential of software development methods: Insights from a work design perspective. *Pacific Asia Journal of the Association for Information Systems*, 9(4), 65-96.
- Kniberg, H., & Skarin, M. (2010). *Kanban and Scrum: Making the Most of Both*. C4Media.

- Lalsing, V., Kishnah, S., & Pudaruth, S. (2012). People factors in agile software development and project management. *International Journal of Software Engineering & Applications*, 3(1), 117-137.
- Lazwanthi, M. R. R., Alsadoon, A., Prasad, P. W. C., Sager, S., & Elchouemi, A. (2016). Cultural impact on agile projects: Universal agile culture model (UACM). *Proceedings of 7th International Conference on Information and Communication Systems 2016*. Irbid, Jordan: IEEE.
- Leewis, S., Smit, K., & van Meerten, J. (2021). An explorative dive into decision rights and governance of blockchain: A literature review and empirical study. *Pacific Asia Journal of the Association for Information Systems*, 13(3), 25-56.
- Li, Y., Hsu, J. S. C., Sun, H., & Parolia, N. (2022). Team leadership and diversity management in information systems development project teams. *Pacific Asia Journal of the Association for Information Systems*, 14(5), 1-29.
- Liechti, O., Pasquier, J., & Reis, R. (2017). Supporting agile teams with a test analytics platform: A case study. *Proceedings of IEEE/ACM 12th International Workshop on Automation of Software Testing 2017*. Buenos Aires, Argentina: IEEE.
- MacGregor, E., Hsieh, Y., & Kruchten, P. (2005). Cultural patterns in software process mishaps. *ACM SIGSOFT Software Engineering Notes*, 30(4), 1-5.
- Marchenko, A., & Abrahamsson, P. (2008). Scrum in a multiproject environment: An ethnographically-inspired case study on the adoption challenges. *Proceedings of Agile 2008 Conference*. Toronto, Canada: IEEE.
- McCormick, K., (1996). Japanese engineers as corporate salary-men. In P. Meiksins, & C. Smith (Eds.), *Engineering Labour: Technical Workers in Comparative Perspective* (pp. 132-163). London: Verso.
- Merhi, M. I. (2021). Multi-country analysis of e-commerce adoption: The impact of national culture and economic development. *Pacific Asia Journal of the Association for Information Systems*, 13(3), 86-108.
- Porschen, S. (2012). Management of the informal by cooperative transfer of experience. In F. Böhle, M. Bürgermeister, & S. Porschen (Eds.), *Innovation Management by Promoting the Informal: Artistic, Experience-based, Playful* (pp. 105-142). Berlin, Heidelberg: Springer.
- Poth, A., Sasabe, S., Mas, A., & Mesquida, A. L. (2019). Lean and agile software process improvement in traditional and agile environments. *Journal of Software: Evolution and Process*, 31(1), e1986.
- Radhakrishnan, J., Gupta, S., & Prashar, S. (2022). Understanding organizations' artificial intelligence journey: A qualitative approach. *Pacific Asia Journal of the Association for Information Systems*, 14(6), 43-77.
- Schein, E. H. (2010). *Organizational Culture and Leadership* (4th ed.). Jossey-Bass: San Francisco, CA.
- Senapathi, M., & Srinivasan, A. (2012). Understanding post-adoptive agile usage: An exploratory cross-case analysis. *Journal of Systems and Software*, 85(6), 1255-1268.
- Seppänen, L., & Poutanen, S. (2020). Cultural transition in the sharing economy: Introducing platform work with activity concepts. In S. Poutanen, A. Kovalainen, & P. Rouvinen (Eds.), *Digital Work and the Platform Economy: Understanding Tasks, Skills and Capabilities in the New Era* (pp. 183-202). London & New York: Routledge.
- Siakas, K. V., & Siakas, E. (2007). The agile professional culture: A source of agile quality. *Software Process: Improvement and Practice*, 12(6), 597-610.

- Simmons, L. L., Mukhopadhyay, S., Conlon, S., & Yang, J. (2011). A computer aided content analysis of online reviews. *Journal of Computer Information Systems*, 52(1), 43-55.
- Spinuzzi, C. (2015). Toward a typology of activities: Understanding internal contradictions in multiperspectival activities. *Journal of Business and Technical Communication*, 29(1), 3-35.
- Takeuchi, H., & Nonaka, I. (1986). The new product development game. *Harvard Business Review*, 64(1), 137-146.
- Tolfo, C., & Wazlawick, R. S. (2008). The influence of organizational culture on the adoption of ExtremeProgramming. *Journal of Systems and Software*, 81(11), 1955-1967.
- Tolfo, C., Wazlawick, R. S., Ferreira, M. G. G., & Forcellini, F. (2011). Agile methods and organizational culture: Reflections about cultural levels. *Journal of Software Maintenance and Evolution: Research and Practice*, 23(6), 423-441.
- Wang, X., Conboy, K., & Cawley, O. (2012). "Leagile" software development: An experience report analysis of the application of lean approaches in agile software development. *Journal of Systems and Software*, 85(6), 1287-1299.
- White, M. D., & Marsh, E. E. (2006). Content analysis: A flexible methodology. *Library Trends*, 55(1), 22-45.
- Yauch, C. A., & Adkins, K. (2004). Effects of cooperative and competitive incentives on agility, quality, and speed in an experimental setting. *Human Factors and Ergonomics in Manufacturing & Service Industries*, 14(4), 403-413.
- Yin, R. K. (1994). *Case Study Research Design and Methods: Applied Social Research and Methods Series*. Second edition. Sage Publications: Thousand Oaks, California.



## Appendix A – Interview Themes in Finland

Interview Themes in Finland
<b>Background questions</b>
What is your education? Would you tell me shortly about your working history? How long have you worked for this company? How long have you worked in your current team?
Would you describe your work? What is your role in the team? Would you describe your normal day at work?
Would you describe your team? How does your team work?
<b>Applying methodology</b>
How well do you know agile methods? How well would you estimate they are known in the organization as a whole?
For how long do you have experience on agile methods, in your company and all in all? What kind of projects have you worked for that have not been realized through agile methods, in this company or in general?
In which different roles have you worked within agile projects?
In which ways have the agile methods been applied?
<b>Agile transformation and its impacts</b>
In which ways have agile methods changed your way of work? What tasks have you done and have not done after applying agile methods? Have you done something differently after applying agile methods?
Why where agile methods applied, what was the motivation of change? Whose initiative was it to start applying agile methods?
How was the transformation realized?
Can you describe the time before and after the implementation of agile methods, what changed?
<b>Impact of agile methods on different aspects of work and teamwork</b>
Which things have been most important to well-being at work when agile methods have been applied?
How has applying agile methods shown in workload and stress experienced at work?
Have 1) meaningfulness, 2) fluency and 3) productivity of your work changed after applying agile methods?
What impact do agile methods have on an individual worker's possibilities to influence his or her own work?
How do agile methods impact on goals set for or by team members?
Do agile methods change giving and getting feedback in your own and in the team's work?
How is starting to use agile methods shown in the competence level of team members?
Do 1) working community, 2) sharing of knowledge, 3) interaction and 4) taking responsibility change when implementing agile methods?
Do agile methods have an impact on leadership and the supervisor's role?
Have agile methods had an impact on the development of work?
What kind of impact do agile methods have on the customer relations?
What kind of effect do the agile methods have on the organization as a whole?
What is the essence (the most important thing in) of agility in your opinion? How would you describe a team that is working in agile way?
<b>Experiences of agile methods</b>
Are you happy with the current ways of working? What is working best? What is the biggest challenge?
Is your development work iterative?
Is your development work incremental?
Are tasks being prioritized? Are the most important features being delivered first?
How are agile principles realized in practice in your opinion?
How has personnel in different levels been committed to agile methods?
Which things are better or worse after applying agile methods?
What kind of thoughts and feelings did applying agile methods provoked in you and your colleagues? What was the reason behind them in your opinion?

## Appendix B – Interview Themes in Japan

Interview Themes in Japan
How do you decide members of a team? What kind of roles do they play? Are they fixed members to some extent?
How do you divide the roles of managers and Scrum Masters if you have them? Who has the best technical capacity? Who has the best management capacity? What do they do when the project faces problems? How has the role of managers changed after agile method were adopted?
How do you decide the concrete method of design and coding? Is it decided by the management, by teams or by individual engineers? If an engineer finds a better way of coding for example, is it possible for him or her to follow it? Do you require engineers to write extensive documentation while coding?
Who decides the deadline of the work? How is it determined? Can members of a team express their opinion, e.g., it is too short?
Is it encouraged to ask other members for help when a member faces problems? Is it actually common to ask someone? From whom do they usually ask? Do they actually hesitate to do so because the idea of independent engineers has been highly admired?
Do you share the problems among the team members? How? Are engineers encouraged to help others solve their problems even when they are not asked to do so? Is “helping others” included in the list of personnel evaluation?
How long do they usually work by alone without any interruption by others? How do you evaluate this number of hours? How long do they usually work in a day? Do you have any specific measures to ensure enough hours to work alone?
What kind of person is the most admired one in a team? Who is it?
Do you try to improve and standardize the work process based on the retrospective?
What factors do you emphasize in the personnel evaluation? What is the most important motivator for engineers? How do you decide the basic pay of engineers?

## Appendix C – Example Quotations from the Themes

Example Quotations from the Themes	
Themes	Example quotations
Power distance: Discretion	Agile practices require highly talented developers... in principle, who self-organize independent teams. However, our teams have not obtained such highly talented developers yet. Thus, our leaders have tried hard to fill the gap between the actual amount of skills and skills needed. (Company A, Leader).
Power distance: Reliance on the leaders	It is our company's new strategy to change leadership style. Previously, leaders were encouraged to force teams to adopt particular developing and reporting methods, which killed the developers' proactiveness. So, we encouraged leaders to coach rather than lead the team members to enhance their proactiveness. (Company C, Leader)
Uncertainty avoidance	After all, we realized good architects are badly needed for agile practices. If we do not have good architects in a project, we produce a lot of bugs in the software. This is the biggest reason why we have failed to fully adopt agile practices. (Company E, Leader)

## About the Authors

**Marja Käsälä** is Specialized Researcher at the Finnish Institute of Occupational Health, Helsinki, Finland. Her research interests include digitalization, new forms of work, platform work and work-life balance. She obtained her D.Sc. Econ. at Turku School of Economics, University of Turku, Finland. She has published nationally and internationally and applied both qualitative and quantitative methods.

**Norio Tokumaru** is a Professor of Economics at the Faculty of Policy Studies at Kansai University, Osaka, Japan. His research interests include innovation policy, the comparative political economy of advanced capitalist economies, and latecomer development in Asia. He obtained his PhD in Economics at Kyoto University, Japan. He has published books and articles on the management of innovation and HRM practices in Japan, South Asia, and East Asia from evolutionary and comparative institutional perspectives.

Copyright © 2023 by the Association for Information Systems. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page. Copyright for components of this work owned by others than the Association for Information Systems must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or fee. Request permission to publish from: AIS Administrative Office, P.O. Box 2712 Atlanta, GA, 30301-2712 Attn: Reprints, or via email from [publications@aisnet.org](mailto:publications@aisnet.org).