IRISH JOURNAL OF MANAGEMENT sciendo

Irish Journal of Management • 42(2) • 2023 • 160-173
DOI: 10.2478/ijm-2023-0010

Irish Journal of **Management**

Deprojectification of agile: The new orthodoxy of long-term product teams

Research Article

Torstein Nesheim*

Centre for Applied Research at NHH, 5045 Bergen, Norway

Abstract: Agile management thinking has influenced organisations in several ways, but the focus of this paper is on the deprojectification of agile software development. The new agile orthodoxy has been promoted by popular books, in blogs and by Spotify and other firms: agile teams should be aligned to products or domains (not projects); cover development, maintenance and operations (to reduce handover problems); and have a long-term character (to build trust, identity and domain knowledge). We first describe and contextualise the new agile orthodoxy and then, based on an empirical study of two different "agile cases," we describe and analyse how employees and middle managers perceived the rationale behind the change to long-term teams, how they understood teams as the basic unit of the structure, their perceptions of the resource and disciplinary dimensions of the organisation and the decline of projects and project managers.

Keywords: *Agile; Deprojectification; Teams; Organisation structure; Innovation*

© Sciendo

INTRODUCTION

Agile management and organisational approaches have influenced organisations in various ways. The agile concept was initiated specifically for software development, on which it has had a considerable influence (Dyba and Dingsøyr, 2009). Recently, it has also had an impact as a generic concept and set of "recipes" for innovation and change in broader settings and organisations (Puranam and Clement, 2018; Rigby, Sutherland and Noble, 2016). Herein, our focus is on a vital but under-researched development in the agile software development domain: *the rationale behind and challenges related to the shift from (temporary) agile projects to stable product- or stream-aligned teams.* In the popular literature, in blogs and the reported success stories of Spotify, Netflix and Amazon, a new model has been promoted that, based on stable, cross-disciplinary teams, encompasses development, maintenance and operations (Barton, Carey and Charan, 2018; Fowler, 2018; Kelly, 2018; Skelton and Pais, 2019; van Gerwin, 2018). However, the new "orthodoxy" has not been reflected in academic research, where agile software development is, to a large extent, still viewed through a project-logic organisational lens (Conforto, Salum, Amaral, da Silva and de Almeida, 2014; Foss and Klein, 2022; Serrador and Pinto, 2015), with rapid reallocation of human resources seen as an enabling factor (Warley, Williams and Lawler, 2014), and research has not addressed the stability and continuity of agile teams (Stray, Moe and Hoda, 2019; Vestues and Rolland, 2021).

To fill gaps in the current literature, we explored two key research questions: 1) What was the rationale behind the change and 2) what characterises the new non-project-based agile organisation? To answer these research

* *E-mail: Torstein.Nesheim@snf.no*

© 2023 Nesheim, published by Sciendo.
This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 License.

questions, we drew on two case studies of agile software development in banking and insurance. We aimed to analyse and describe how employees and middle managers made sense of and understood the rationale for the change to long-term product teams and how they perceived teams as the basic unit of the wider organisational structure in which the agile philosophy is embedded. The research involved an empirical analysis of continuous (rather than project-based) agile units in the software domain.

Given the emergence of new conventional agile wisdom for organisations (Barton et al., 2018; Kelly, 2018; Skelton and Pais, 2019; van Gerwin, 2018), this empirical study has the potential to add to the knowledge of this emerging organisational phenomenon. Our main contributions are the identification of six dimensions that capture the change from project-based to 'permanent' product-based software development, as well as suggestions for further research.

ORANISING AGILE SOFTWARE DEVELOPMENT

The agile organisational phenomenon involves cultural aspects, mindsets and organisational and management practices. Notable outcomes of this phenomenon have been a great deal of literature for practitioners (comprising "how to" books, articles, blogs, videos and podcasts) and a large body of management and organisation research. In the academic literature, the agile methodology may relate to a) an organisational *ability*, such as the ability to recognise organisational changes in the environment and respond swiftly and efficiently (agility) (Felipe, Roldán and Leal-Rodríguez, 2016; Zitkiene and Diksnys, 2018) and/or b) a set of organisational mechanisms, processes, methods and tools that *promote and enable* such agility (Dyba and Dingsøyr, 2009; Stray, Moe and Hoda, 2019). In this paper, we highlight the latter.

The agile manifesto

A key event in the agile trajectory was the publication of the Agile Manifesto (Beck et al., 2001), which established a set of principles and practices that emphasised individuals and interactions (rather than processes and tools), working products (rather than comprehensive documentation), customer collaboration (rather than contract management) and responding to change (rather than following a plan). In contrast to Waterfall projects, people were now expected to "work in cycles, using frequent, time-boxed iterations that allow regular check-ins with and feedback from their colleagues and their end-product customers" (Krehbiel et al., 2017, p. 90).

Agile projects

The agile way of working has had a profound influence on software development (Dyba and Dingsøyr, 2009) and as a generic management idea beyond the software domain (Madsen, 2020). Software development research has (implicitly or explicitly) emphasised agile a project logic. The evolving practices of cross-disciplinary interaction, iterations, customer collaboration and incremental change contrast with Waterfall projects, which are characterised by excessive formalisation and specifications. However, a project logic still dominates within agile approaches and research on agile software organisations and methods has to a large extent analysed the benefits and challenges of Agile projects, typically contrasting them with Waterfall projects (Foss and Klein, 2022[1]; Conforto et al., 2014; Dybå and Dingsøyr 2009; Felipe et al., 2016; Hobbs and Petit, 2017; Serrador and Pinto, 2015). Studies on agile teams have tended to assume that software development proceeds according to a project logic and a project mindset, generally failing to address the permanence and stability dimensions (Grass, Backmann and Hoegl, 2021; Gren, Torkar and Feldt, 2017; Przybilla, Wieser and Krcmar, 2019; Vishnubhotla, Mendes and Lundberg, 2017). This organisational logic mirrors the ambidexterity approach based on the notion that existing organisations need to combine exploitation (efficiency and fine-tuning) with exploration (creativity and innovation) (O'Reilly and Tushman, 2016; Raisch and Birkinshaw, 2008). These two sets of activities differ and require different mindsets, goals, leadership and competencies. Software development is usually handled by time-bound, autonomous teams that ideally follow agile practices and share an agile mindset. These units are disbanded upon completion of the development project (Hobbs and Petit, 2017; Kelly, 2018; Puranam and Clement, 2018; Serrador and Pinto, 2015). As stated by Puranam and Clement (2018):

1 In their book, Foss and Klein (2022) stated (regarding Spotify), "At the end of the day, Spotify is a relatively traditional, project-based company mixing a proven organizational structure with Agile development ideas" (p. 76).

Instead of marching in bureaucratic formation, they are given freedom to "sprint," i.e., rapidly advance in parallel towards a solution with minimal top-down oversight. Interdependencies between the various teams are managed through frequent meetings and any rework is accepted as part of the iterative process. Upon completion of the immediate task, teams either dissolve or take on a new project. (blogpost).

From agile projects to agile product teams: The new orthodoxy?

The project logic and (implied) temporary mindset have been criticised in the popular agile literature, since such short-sightedness hinders the development of trust and cohesion in teams, requires overly complex governance mechanisms and creates extensive handover challenges between development and operations departments (Kelly, 2018; Skelton and Pais, 2019). Instead, an alternative model has been promoted based on stable, cross-disciplinary teams that encompass development, maintenance and operations and supported by the success stories of Spotify, Netflix, Amazon and ING (Barton, Carey and Charan, 2018; Kelly, 2018; Skelton and Pais, 2019; van Gerwin, 2018).

A typical description of the new form of team is as follows:

> Any organisation, when they first start working with Scrum or other agile frameworks, does so in project teams. These teams stay together for the duration of the project, meet regularly and deliver the project within a short time frame. …By stable agile teams, we refer to cross-functional teams that can carry out an assignment from end to end. This kind of team works together on the same assignment full-time and is not dissolved when the assignment is completed. (Claessen, 2018).

Thus, stable teams as an organisational solution challenge agile a project logic. Kelly (2018) argued that software development has been hindered by the excessive emphasis on a project logic and project management models, which advocate working in time-bound project teams. The problem with a project logic is partly the temporariness of teams (teams disbanded quickly and maintenance and operation outside the responsibility of those teams) and partly the complex governance mechanisms associated with project management. According to Kelly (2018), software development should be regarded as continuous, requiring organisations to be based on continuous, long-term agile teams. Teams should prioritise long-term product or business domains rather than time-bound projects (Kelly, 2018).

Similar ideas were described by Skelton and Pais (2019) and DevOps proponents. Skelton and Pais (2019) stated that teams are the basic operational delivery units in an organisation. They should be diverse and cross-disciplinary, remain stable over time and be kept small to develop trust between members (Skelton and Pais, 2019). The DevOps approach emerged due to the growing friction between software development and operations (Kim, Debois, Willis and Humble, 2016; Wiedemann, Gewald, Wiesche and Krcmar, 2019). Previously, development and operations teams had separate responsibilities, with software releases "thrown over the wall" to operations. The methods did not address the gap between development delivery speed and operational teams' capacity to provide resources and deploy updates. Development and operations should be better coordinated, and teams may be a way to achieve this (Kim et al., 2016; Wiedemann et al., 2019).

Coordination challenges arise at a higher organisational level when several cross-disciplinary agile teams are dependent om each other. Given the focus on cross-disciplinary working in a single team, how should issues of discipline-specific development, knowledge sharing inside a specialised sphere of competence, identification with a discipline and personnel management be handled? De Smet (2018) recommended an "agile matrix" that supplements the team (product) dimension of a structure with a disciplinary or functional manager who can "build up the right capabilities and people, equip them with the skills, tools and standard approaches to deliver functional excellence and ensure that they are deployed to value-creation opportunities, not involved in the day-to-day work of squads" (De Smet, 2018, p. 3). Such an agile matrix resembles a resource matrix, as described in the organisation design literature (Nesheim, 2021).

Empirical studies on agile product teams and DevOps

Some studies have addressed the new "conventional wisdom" in agile software development. For example, Gerster, Brenner and Dremel (2020) emphasised product orientation and stable teams that embrace development and operations as key agile features and structures. This approach contrasts with a traditional functionally based "silo" approach to organisation. However, the contrast with and changes from an agile project approach are mentioned briefly and are not discussed by the authos. The DevOps approach prompted several empirical studies, most often

regarding information systems. In a review article, Leite, Rocha, Kon, Milojicic and Meirelles (2019) identified six fundamental concepts and seven key tools for DevOps, none of which capture the product versus project dimension. However, the authors briefly discussed three organisational alternatives for DevOps deployment: collaboration across departments, cross-functional product teams and DevOps teams. Based on a qualitative study, Wiedemann, Wiesche, Gewald and Krcmar (2020) found three factors that contribute to IT and business alignment: individual componentisation, integrated responsibility and multi-disciplinary knowledge.

Studies have considered the "solutions" and challenges of agile product teams and Dev-Ops ideas and practices, but few studies have focused on the product versus project dimensions and the stability versus temporality of teams. Pulakos, Kantrowitz and Scheider (2019), exceptionally, argued that stability "provides a secure foundation that helps people stay focused on performance during disruptive change, rather than being worried and distracted about what may happen" (p. 309). They found that stability, rightsized teamwork and relentless course correction play especially important roles in fostering organisational agility and resilience, which in turn lead to substantially higher organisational financial performance. However, they did not explicitly address the stability of team membership in their study.

DATA AND METHODS

Given the nature of the research questions, we chose a qualitative methodology to capture the meanings, perceptions and intentions of the actors involved in agile organization and designed an empirical study of two cases/ organizations (Alpha and Beta), combining exploratory and comparative elements to examine their different forms of agile organisation (cf. Yin, 2013). The companies operated in two sectors (banking and insurance), in which software-based processes and services are known to be important. We gained access to Alpha (a bank) in February 2021, conducted two background interviews with a key informant and obtained further information from organisation charts, reports and presentations. The Development Division had recently been reorganised into cross-functional teams. Based on purposeful sampling, we drew primary data from 19 semi-structured interviews with respondents in different roles, which we conducted on Zoom and Teams in March 2020.

Based on theoretical sampling principles (Eisenhardt, 1989), we sought a second case that had both similarities to and differences from the first case. In September 2021, we gained access to Beta (an insurance company), which had also conducted agile implementation in part of the organisation. The main difference between the two cases was that an entire Alpha division was based on an agile team, whereas Beta had three locally initiated "agile islands" across two divisions. Thus, the two cases resemble polar opposites (Eisenhardt, 1989) regarding the prevalence of agile teams in the affected divisions. The empirical settings will be discussed in more detail prior to the empirical findings. In Beta, interviews with 12 respondents in different roles were conducted in November and December 2021.

In both firms, respondents included middle managers, people who held specialised agile positions as product owners, agile leads or technical leads and other members of product teams. We based the interview guide for Alpha on the relevant literature and background information on the division in question. The interviews covered the respondents' backgrounds and expertise, how they perceived the change to agile teams, key aspects and challenges of agile teams, coordination across teams, employee development and specialised competencies. We performed an initial analysis of the data before the Beta interviews. Thus, the interview guide for the Beta interviews was more focused and emphasised the project versus product dimensions of the change from a project to a product mindset.

We conducted the data analysis in three stages. First, we analysed the Alpha data to identify, classify and analyse the respondents' utterances, taking an open, interpretive approach (Schwandt, 1994) to capture respondents' perceptions of the key issues and challenges. For each interview, we coded all relevant utterances (first order), assigned each first-order code order to a subtheme (second order) and then assigned subthemes to themes (third order). Second, based on the findings from Alpha and the refined interview guide, we conducted a similar but less open analysis of the Beta data. Finally, in the third stage, we compared the two cases and identified six subthemes (see Results section, below). The data analysis and interpretation processes took over a year to complete and had iterative aspects, enabling us to move back and forth vertically (between orders) and horizontally (between respondents and between categories) in each case, as well as between the two cases. The Appendix provides an overview of the respondents and an example of the coding of one of the subthemes.

THE EMPIRICAL SETTING

Alpha is a well-established Norwegian bank with 700+ employees. The Development Division has approximately 200 employees, including ICT developers, architects, testers, interaction designers, risk and safety specialists and employees with management qualifications. From 2015 to 2019, the division structure was gradually changed into a standardised organisation based on cross-disciplinary, long-term teams as operational units. In addition to establishing the team-based structure, the bank simplified governance mechanisms, introduced initiatives to make the division culture more commercial and customer-focused; and increased the use of agile methods, roles and tools, such as design thinking.

Beta is a Scandinavian insurance company with 2000+ employees. In the Norwegian part of the organisation, three agile teams had developed incrementally in the domains of digital shops, digital services and digital claims. Participants in these teams had similar competencies and roles as the employees in the Innovation division in Alpha.[2]

Respondents in both firms were familiar with the popular agile literature and were inspired by how firms such as ING, Netflix and Spotify developed their organisation structures and used agile methods and mindsets. However, they pointed out that they did not copy or imitate a specific solution or model, instead using external sources as inputs to the internal change processes in both organisations.

RESULTS

The problem with development projects (T1)

Before the changes (2015 in Alpha, 2017 in Beta), processes and product development in both firms were typically organised according to time-bound projects that were separate from operations. Informants across the firms claimed that this structure had many disadvantages, one of which was related to the handover of solutions from projects to operations. Participants in development projects were not involved in their operational implementation; there was often minimal communication and feedback between the two functions and a lack of long-term ownership of solutions.

Another issue was the detailed control, specification and documentation required for development projects. Respondents perceived no support for autonomy and creativity, which reduced the speed and degree of innovation. Alpha 16 provided an illustrative statement, as follows:

> There was a low degree of autonomy in project teams . . . [In practice], development took place through specifications, templates . . . and decisions taken outside teams to a large extent. It was very Waterfall-oriented . . . ownership of the product was low and the understanding of what the teams built tended to be low.

Towards long-term product teams (T2)

Reorganisation into agile product teams started in 2015 in Alpha and in 2017 in Beta. In Alpha, the structure of the entire Development Division was gradually changed into an organisation based on cross-disciplinary, long-term teams as operational units. The new team-based structure was supported by a simplified governance structure, initiatives to foster a more commercial and customer-focused culture and the use of agile roles and tools, such as design thinking. In contrast, in Beta, product teams emerged more selectively and mainly as three “agile islands”: digital sales, digital service and digital claims.

In both organisations, products were the foundation of the new teams. Each team now “owned” a product or domain that, rather than being time-bound, had a measure of permanence. The organisational logic was that, if a product or domain existed, it should be supported by a team. As one of the respondents stated: “For example, we will always have payments and transactions, so we will always need clever people in these areas. When you have lasting domains, it is very sensible to have lasting teams” (Alpha 14).

When organising by product, respondents perceived that a different dynamic emerged. As one respondent said:

2 Both Alfa and Beta had a number of external consultants (especially developers) in their teams besides the employees. We do not address issues related to employees versus consultants in this paper.

> You must both operate it and maintain it (in the same team). Now, the logic is: If you build it, you own it, you run it, you kill it. Developers need to think about how to develop a solution that the team can operate and maintain. Employees need a more cross-disciplinary approach . . . and to think more in terms of the life course of a solution. You need to avoid shortcuts that accelerate development but increase the cost of operations. (Alpha 8)

Although the change from distinct project and operations units to a product-based organisation was easily recognised in retrospect, the product teams tended to emerge gradually. This incremental process was especially prominent in the case of Beta, where the changes (to three product teams) took place locally and not according to a division-level plan, in contrast to Alpha. The process was explained by a Beta informant as follows:

> The change from development projects to products did not take place overnight, but more gradually. We recognised that it was more sensible to work with a long-term horizon in the teams in question . . . I don't remember the exact date, but we gradually started to use the term "product team" rather than "project team" . . . It took a long time for parts of the organisation to recognise this change. (Beta 8)

The operational units: Stable, cross-disciplinary teams (T3)

The teams were similar in the two organisations. Typically, a team was cross-disciplinary and included developers, architects, UX designers, tech leads and product owners. With few exceptions, each employee was a member of a long-term team. Thus, the organisation of the Development Division in Alfa and the "agile islands" in Beta differed from project-based structures, temporary teams and organisations in which employees were members of several (stable) teams. A critical dilemma in team-based organisations is that of stable versus flexible team membership. In both firms, respondents emphasised the advantages of team stability, claiming that trust and relationship-building among team members are vital. As one respondent stated:

> Relations are very important. If you look at teams (with membership changes) . . . they slow down the work . . . Members of a team who have worked together over time have come to know each other, have worked on processes and learned how to work smarter. They are much more productive than a team that has experienced changes [in membership]. (Alpha 14)

The other main driver of membership stability was domain competence, exemplified by the following perception:

> You build knowledge of the domain. For example, the loans team needs to understand the settings in which customers must borrow, how our credit model functions, what data we need and how we may obtain that data. They develop domain knowledge and a cooperative climate that we intend to conserve as long as possible. (Alpha 6)

Thus, in a team with stable membership, employees perceived that they had a long-term, safe framework for their work, came to know their colleagues well and developed in-depth domain-specific knowledge. They were allowed to work consistently in one area without having to switch between tasks in different teams. At the team level, perceived advantages were the decreased need for onboarding and other processes due to frequent changes in team membership. Teams that experienced many such changes were perceived to be less efficient.

Although one-team-per-employee was clearly preferred, this was not always feasible. A lack of specialised technical competencies or cases in which the amount of work was less than full-time could result in some employees working across teams.

Regarding team membership, although all informants desired stability, many experienced dilemmas. They perceived it as important for teams not to get stuck in a pattern because, sometimes, the team structure was less than ideal and needed to be changed. In addition, although the product or domain might be long-term in nature, a need for capacity changes or special competencies could arise during the lifetime of the team. Respondents perceived that, in some cases, teams could become "too strong" and create barriers to necessary changes in membership.

The autonomy of teams (T4)

The informants in both organizations claimed that management tended to emphasise the increased autonomy of teams and give team members more responsibility than before. However, the conditions for such autonomy differed. In Alpha, team autonomy was supported by compatible division-wide changes in governance structures, cultural initiatives and agile ways of working. In Beta, the three product teams were embedded in a more complex organisational setting, coexisting with other organisational and hierarchical structures.

One Alpha informant described the new teams as follows:

> It is now more up to the team (to make decisions). An increased level of autonomy is very positive. From my point of view, the biggest effect is that each member is expected to own what he or she has built from the concept phase to production and use by the customer . . . They perceive that they have created something for the customer, in contrast to before, when they were given an instruction and executed it. (Alpha 17)

In both organisations, teamwork was coordinated by a product owner who was responsible for dealing with backlogs, activity coordination and relations with "neighbour" teams and other parts of the organisation. Although this was a key role, informants emphasised that the position holders should not micromanage processes or team members, but instead create conditions to foster creativity, empowerment and cooperation in their teams. Specific agile roles (such as agile coach and tech lead) and agile methods (e.g., Scrum, digital boards and design thinking) were used extensively in both organisations.

The teams were perceived to have greater autonomy and to provide better solutions through working across disciplines. A vital question in both Alpha and Beta was whether all relevant disciplines were involved. Many respondents expressed a preference for involving disciplines other than ICT to align solutions better with the relevant markets and customers. In both firms, respondents pointed out that there was still some way to go:

> Simply put, we are cross-disciplinary ICT and are not close enough to the business. Some teams lack people with commercial and business knowledge. Thus, there is sometimes a distance between the teams and the core business, and we don't really understand each other. (Alpha 19)

Supplementary structures for personnel management and knowledge sharing (T5)

In each "agile" team in the two organisations, employees' attention was directed towards *one* team and the interaction and problem solving within that team. This was the task dimension of the organisation related to a product or a domain. Given the focus on cross-disciplinary teams, two other questions arose during the interviews:

- How was the resource dimension handled, and who was responsible for personnel management?
- Despite being a member of a cross-disciplinary team, an employee also has specialised competencies in, for example, development, design or architecture. How can specialised competencies be shared and developed in an organisation that emphasises cross-disciplinarity teams?

In Alpha, a product- and team-based structure was established in 2019. Initially, the previous discipline-based departments were maintained alongside the new teams, so, for some time, the organisation resembled a matrix structure, with employees having formal dual attachments to their teams and the departments they reported to. This structure was not considered optimal. The respondents claimed that the dual attachments tended to pull employees in different directions and result in a lack of alignment between the two entities' interests: "You were drawn in different directions, with different priorities, different managers, different roles and expectations" (Alpha 7). There were also challenges related to identification and personal management. As one respondent explained:

> Developers felt more at home in the Systems Development Department. Teams depend on psychological safety and a sense of community, but the department (not the team) was the main arena for meetings and other social contact . . . [In a matrix], the personal manager cannot observe your work and has limited knowledge of what you do, what tasks you work on and how to prioritise. (Alpha 6)

After a while, the departments (and, consequently, the matrix) were dissolved. Instead, coordination was introduced between the teams and the division in the hierarchical structure. At this level (the "tribe" level), which comprised three to six "neighbour" teams, strategic direction, coordination of teams in the tribe and personnel management

were provided for them.

Many respondents mentioned the development and sharing of special competencies among employees across teams. They acknowledged a balancing act between cross-disciplinarity and customer focus in teams on the one hand and the development of specialised, discipline-specific competencies across teams on the other hand: “There is a potential trap if you are not conscious that it may turn out that way and don’t develop your field of expertise” (Alpha 18).

While employee development usually took place during the day-to-day activities of the product teams, learning and knowledge development also occurred outside them through, for example, monthly “learning days.” Managers encouraged employees to interact with other employees with similar competencies across teams, as well as to participate in activities and arenas that involved other competencies. Learning days were open to everyone in the division, but the emphasis was on the largest disciplines. To enable discipline knowledge sharing and development, several guilds or communities of practice were established, each of which had a coordinator with a rather “loose” job description.

As described previously, the Beta organisational context was different and raised the issue of whether secondary structures should supplement the “islands” of cross-disciplinary teams. Most team members belonged to the Technology Division, which had a matrix organisation structure, comprising resource units based on employees with similar competencies, teams, and other task units, such as project units. Thus, the members of the three teams were part of a matrix organisation and reported to both their resource units and their teams. Thus, the resource unit was responsible for knowledge sharing and development within the relevant sphere of expertise. In contrast to Alpha, which had *separate* supplementary structures to handle the resource and knowledge sharing dimensions, the resource units in the Beta matrix structure were responsible for both.

The decline of projects (T6)

In both organisations, informants claimed that development projects were an established tool for organising change and reflecting stability–change–stability cycles. The interviews indicated a high level of change in the organisations, combined with a lower prevalence of projects (as temporary suborganisations) and a reduced focus on project logic and a project mindset as means to facilitate change. These perceptions were evident in both organisations. Agile teams now handled activities that were previously part of development projects. As Beta 6 explained:

> [Activities] that were previously part of a project are now a part of agile teams . . . That’s the big picture. There are still projects alongside agile teams, but a large part of what previously would have been a project has now moved inside an agile team.

In Alpha, informants tended to downplay the importance of projects to an even greater extent. Although they recognised the project manager role and engaged in temporary tasks, these were not regarded as projects. As expressed by one informant: We don’t have projects anymore. (Alpha 16).

Given the different ranges of agile teams in Alpha and Beta, the decline in project organisation and project logic was more extensive in the former. The interviews in both organisations clearly indicated that the project manager role had become less important than the product manager role. Many previous incumbents of the project manager role now work as product managers or team leads but may draw on their experience in agile project methods.

DISCUSSION

In this research, we studied the rationale and challenges of the shift from (temporary) agile projects to stable product- or stream-aligned teams. We studied two organisations that had implemented different scopes of agile teams (an entire department vs. “agile islands”) and exhibited different degrees of deprojectification. The empirical study revealed that middle managers’ and team members’ perceptions largely matched “the problems with projects” highlighted in the popular literature (Fowler, 2018; Kelly, 2018; Skelton and Pais, 2019), such as complex structures and governance mechanisms and a limited (discipline-specific) short-term mindset that hinders a long-term commercial approach.

In both organisations, new, long-term teams were aligned with stable products and cooperated using agile roles, methods and tools. The desire for stability and the descriptions of agile autonomous teams were similar

across the two organisations. The main differences between the two cases (the degree of simultaneous changes in governance and culture and the degree of project decline) were explained by the different prevalence of agile teams in Alpha and Beta.

The advantages of stable teams are long-term relations within teams and domain-specific knowledge. Respondents in the two cases agreed that stable team membership positively influences development and maintenance processes. This finding may accentuate a dilemma in organisations with agile teams. Stability and permanence may hinder necessary change and development, such as downsizing of and changes of membership in teams. There may be difficulties with closely knit teams in the overall restructuring of a team-based organisation. Potential changes may be triggered by developments in markets and technologies, changes in strategic direction or prioritisation tasks between teams. When members perceive high levels of trust and identity in their teams, they may be reluctant to start working in another team (despite managers considering this appropriate). Thus, the strong team structure in these organisations may result in a dilemma between stable team membership on the one hand and learning and flexibility on the other.

Regarding the introduction of supplementary structures to cross-disciplinary teams, the two cases revealed similarities and differences in organisation. Personnel management (and how it relates to task management) and the sharing and development of specialised knowledge were issues for both organisations, but their chosen solutions differed. In Alpha, where the Development Department was transformed into a team-based structure, the initial resource/team matrix was dissolved, contrasting with typical recommendations for agile matrices in the popular literature (De Smet, 2018). In Beta, the three teams were located in an existing matrix structure and each team member had a "home" in a discipline-specific resource unit.

Based on the research findings, we identified a second dilemma for "agile organisations": on the one hand, the new "orthodoxy" recommends cross-functional, agile teams as basic operational units, but on the other hand, it suggests that the mindset and attention to a specific domain and team should be balanced with a) the need to belong to and share knowledge within a discipline (or a set of specialised competencies) and b) personnel management and responsibility for human resources in the organisation. Although some arguments have supported an "agile matrix" (De Smet, 2018) based on specialised resources and cross-functional teams, this study suggests that other important considerations should be taken into account. There may be a "history" regarding the resource dimension, which was "too strong" in Alpha but in Beta related to an existing structure in which teams were placed. The solution chosen by Alpha (the "tribe" as a level between team and department, with a specific role for the resource dimension outside a Matrixes) seemed to avoid some of the complexities of pure resources/teams matrices.

Although this empirical study targeted the software domain, the insights of agile organisations may be of broader relevance to organisations. Software and ICT have gradually become more integrated into organisations (Kettunen and Laanti, 2017). In addition, agile methods and principles are being diffused across functions, departments, organisations and sectors, assuming the characteristics of generic managerial ideas (Madsen, 2020). One domain of interest is organisational structure, where both the agile literature and the generic research on teams advocate stable, cross-disciplinary teams as the operational building blocks of an organisation. Since this is obviously a popular management idea (cf. Cram and Newell, 2016), it may be widely applied and imitated. Based on this observation and the findings of the study, two questions arise: 1) Under what conditions are such a primary structure based on cross-disciplinary teams appropriate? 2) Given the decision to apply cross-functional teams as building blocks, what are the perceived "needs" (e.g., for identity, knowledge sharing, personnel management and supplementary structures) and how can they be met by building these dimensions into the organisation structure? This study enabled us to identify several relevant considerations ("history," timing and existing structures), and future research may combine insights from the "agile domain" with more generic studies of matrix and other two-dimensional organisations.

A second research theme related to the mechanisms by which communities of practice promote domain- or discipline-specific knowledge sharing across cross-functional teams. Although such mechanisms have been analysed both inside (Smite, Moe, Levinta and Floryan, 2018) and outside (Nesheim, Olsen and Tobiassen, 2011) the software sphere, few studies have so far considered the dilemma identified herein: how to promote discipline-specific knowledge development when a cross-disciplinary mindset prevails (Annosi, Foss and Martini, 2020; Bredin and Edberg, 2020). Based on this study, future studies may examine the degree of formalisation (of roles, etc.) in such processes, considering different emphases across development phases in an organisation.

Finally, the study provides insight into the vital question of the significance of projects, project management and project managers in organisations. Although the project management literature advocates a "projectification"

process and frequent reallocation of human resources (Jacobsson and Jalocha, 2021; Lundin et al., 2015; Maylor and Turkalainen, 2019), we have identified “deprojectification” mechanisms that may be relevant both inside and outside software-based departments and organisations. To handle development and an increasing rate of change, cross-functional, stable teams have been established in the software sphere. Such agile teams take on the task of continuous improvement, meaning that many change initiatives are now organised outside development and change projects. With the diffusion of agility and stable teams beyond the software sphere, the question of “deprojectification” and its accompanying mechanisms and processes may be raised on a broader scale. Thus, agile influences may impact the degree of “projectification” of an organisation in terms of time-limitation, the definition of tasks and the uniqueness of team membership (cf. Bakker, 2010). How does the new agile orthodoxy emphasising stable teams influence and challenge the prevalence of project logic and how should firms organise for innovation and change? If there is growth in an agile organisation (“agile at scale”), what is the impact of that growth on the demand for project managers and to what extent must competencies be redefined to mirror new challenges of organisational development and change?

CONCLUSION

Agile methods and concepts represent a collection of influential management ideas. In this article, we have examined a “new orthodoxy” that has taken root in the agile ICT/software domain (Kelly, 2018; Skelton and Pais, 2019)—the idea of organisations based on autonomous, stable, cross-functional, product-focused teams, which contrasts with project logic and recommendations for rapid reallocation of resources and is perceived to enable better coordination between development, maintenance and operations. We have contributed to the literature by documenting changes from project to product teams and by identifying and examining six aspects of such agile structures in an empirical study. Based on this study, we also suggest several themes for future research.

The study has several limitations in terms of the number of organisations and the methods employed. Since we studied only two cases in Norway, the findings cannot be generalised to all Norwegian firms or firms in other countries. The data collection was limited to one point in time in each case; hence, a longitudinal study of the cases using real-time interviews would be desirable.

This study may inform future research on agile organisations in the software sphere and agile influences beyond this “home” domain. In the former, possible research themes include the stable membership/flexibility dilemma, how cross-disciplinary teams may coexist with the sharing and development of specialised knowledge, and the coexistence of product (and other long-term) teams and development projects. These issues are also relevant for organisations that apply agile methods and organisational principles outside the software domain. The coexistence of autonomous teams with other structures, principles and logics are particularly relevant issues. The diffusion of agile methods and principles in an organisation may challenge both the influence of the existing hierarchy and the power and influence of project managers and project logics (vs. stable units).

Torstein Nesheim is a Senior researcher / Professor at Centre for Applied Research at NHH (Norway). His main research interests are organizational structure, agile organization, project management and non-standard work arrangements. He can be reached at: Torstein.Nesheim@snf.no

References

Annosi, M.C., Foss, N. and Martini, A. (2020). When Agile harms learning and innovation: (and what can be done about it). *California Management Review*, *63*(1), 61–80.

Bakker, R.M. (2010). Taking stock of temporary organizational forms: A systematic review and research arena. *International Journal of Management Review*, 12 (4): 466–468.

Beck K., Beedle, M., van Bennekum, A., Cockburn, A., Fowler, M., Grenning, J., Highsmith, J.A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaberm K., Sutherland, J., and Thomas, D. (2001). *Manifesto for Agile software development*. http://agilemanifesto.org/. Accessed March 25, 2023.

Bredin, K. and Engberg, C. (2020). *Agile but Fragile? Practices for Sustained Disciplinary*

Expertise in Project-Based Oranizations. Working paper. Linkøping University.

Barton, D., Carey, D. and Charan, R. (2018). One bank's Agile experiment: How ING revamped its retail operation. *Harvard Business Review*, 96 (2): 59–61.

Claessen, B. (2018). *From project teams to stable agile teams*. https://www.organizeagile.com/update/from-project-teams-to-stable-agile-teams/. Accessed March 25, 2023.

Conforto, D.C., Salum, F., Amaral, D.C., da Silva, S.L. and de Almeida, L. (2014). Can agile project management be adopted by industries other than software development? *Project Management Journal*, *45*(3), 21–34.

Cram, M.A. and S. Newell (2015). Mindful revolution or mindless trend? Examining agile development as a management fashion. *European Journal of Information Systems, 25*(2) 154–165.

De Smet, A. (2018). The Agile manager: Who manages in an agile organization (and what exactly do they do)? *McKinsey Quarterly*, July: 1-6.

Dybå, T. and Dingsøyr, T. (2009). What do we know about agile software development? *IEEE Software,* 26 (5); 6–9.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, *14* (4), 532–550.

Felipe, C.M., Roldán, J.L. and Leal-Rodríguez, A.L. (2016). An explanatory and predictive model for organizational agility. *Journal of Business Research*, *69*(10), 4624–4631.

Foss, N.J. and Klein, P. (2022). *Why Managers Matter: The Perils of the Bossless Company.* New York, NY: Public Affairs.

Fowler, M. (2018). *Products over projects.* https://martinfowler.com/articles/products-over projects.html. Accessed March 25, 2023.

Gerster, G., Brenner, W. and Dremel, C. (2020). *How Enterprises Adopt Agile Forms of Organizational Design: A Multiple-Case Study*. ACM SIGMIS Database. https://www.researchgate.net/publication/338731956_How_Enterprises_Adopt_Agile_Forms_of_Organizational_Design_A_Multiple-Case_Study/link/5e3d4251a6fdccd9658e7a4c/download. Accessed March 25, 2023.

Grass, A., Backmann, J. and Hoegl, M. (2020) From empowerment dynamics to team adaptability: Exploring and conceptualizing the continuous agile team innovation process. *Journal of Product Innovation Management, 37*(4), 324–351.

Gren, L., Torkar, R. and Feldt, R. (2019). Group development and group maturity when building agile teams: A qualitative and quantitative investigation at eight large companies. *The Journal of Systems and Software*, Vol.124, 104–119.

Hobbs, B. and Petit, Y. (2017). Agile methods on large projects in large organizations. *Project Management Journal, 48*(3), 3–19.

Jacobsson, M. and Jalucha, B. (2021). Four images of projectification: An integrative view. *International Journal of Managing Projects in Business*, *14*(7), 1583–1604.

Kelly, A. (2018). *Project Myopia: Why Projects Damage Software: No-projects.* Manchester: Allan Kelly Associates.

Kettunen, J. and Laanti, M. (2017). Future software organizations – agile goals and roles. *European Journal of Futures Research*, *5*(16), 1–16.

Kim, G., Debois, P., Willis, J. and Humble, J. (2016). *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. Portland, OR: IT Revolution.

Kniberg, H. and Ivarsson, A. (2012). *Scaling Agile @ Spotify with tribes, squads, chapters and guilds.* https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf. Accessed March 25, 2023.

Krehbiel, T.H., Salzarulo, P-A., Cosmah, M., Forren, J., Gannod, G., Havelka, D., Hulshult, A. and Merhout, J. (2017). Agile manifesto for teaching and learning. *The Journal of Effective Teaching*, *17*(2), 90–111.

Leite, L., Rocha C., Kon, F., Milojicic, D. and Meirelles, P. (2020). A survey of DevOps concepts and challenges. *ACM Computing Surveys*, 52(6), 1–35.

Lundin, R.A., Arvidsson, N., Brady, T., Ekstedt, E., Midler, C. and Sydow, J. (2015). *Managing and Working in Project Society*. Cambridge: Cambridge University Press.

Madsen, D. (2020).. "The Evolutionary Trajectory of the Agile Concept Viewed from a Management Fashion Perspective", *Social Science 9* (69): 1-22.

Maylor, H. and Turkulainen, V. (2021). The concept of organizational projectification: Past, present and beyond?" *International Journal of Managing Projects in Business*, *12*(3), 565–577.

Nesheim, T. (2021). Exploring the resource manager role in a project-based organization. *International Journal of Managing Projects in Business, 14*(7),1626–1641.

Nesheim, T., Olsen, K.M. and Tobiassen, A. (2011). Knowledge communities in matrix-like organizations: Managing knowledge towards application. *Journal of Knowledge Management*, *15*(5), 836–850.

O'Reilly III, C.A and Tushman, M.A. (2016). *Lead and disrupt. How to solve the innovator's dilemma.* Stanford, CA: Stanford Business Books.

Przybilla, L., Wiesche, M. and Krcmar, H. (2019). *Emergent Leadership in Agile Teams–an Initial Exploration*. Conference proceedings. SIGMIS-CPR, 19, June 20–22, 2019, Nashville, TN, USA

Pulakos, E.D., Kantrowitz, T. and Schneider, B. (2019). What leads to organizational agility: It's not what you think. *Consulting Psychology Journal: Practice and Research, 71*(4), 305–320.

Puranam, P. and Clement, J. (2018). *Why agile may be fragile*. https://knowledge.insead.edu/blog/insead-blog/why-agile-may-be-fragile-10201. Accessed March 25, 2023.

Raisch, S. and Birkenshaw, J. (2008). Organizational ambidexterity: Antecedents, outcomes, and moderators. *Journal of Management,* 34 (3): 379–405.

Rigby, K.R., Sutherland, J. and Noble. A (2018). Agile at Scale. *Harvard Business Review*. May–June: 88-96.

Schwandt, T.B. (1994). Constructivist, interpretive approaches to human inquiry. In N.K. Denzin and Y.S. Lincoln (Eds.), *Handbook of Qualitative Research* (pp. 118–137). Thousand Oaks: Sage Publications, Inc.

Serrador, P. and Pinto, JK. (2015). Does agile work?—A quantitative analysis of agile project success. *International Journal of Project Management 33*(5), 1040–51.

Skelton, M. and Pais, M. (2019). *Team Topologies.* Portland, OR: IT Revolution.

Smite, D., Moe, N.B., Levinta, G. and Floryan, M. (2018). How to succeed with knowledge sharing in a large-scale Agile organization. *IEEE Software*, 36 (2): 51–57.

Stray, V., Moe, N.B. and Hoda, R. (2019). Autonomous agile teams: Challenges and future directions for research. 19th International Conference on Agile Software Development. Conference Proceedings of XP18 Companion, Porto, Portugal.

Van Gerwin, M. (2018). *From project teams to stable Agile teams*. Blogpost, https://medium.com/organize-agile/from-project-teams-to-stable-agile-teams-5934c271a8fc. Accessed March 25, 2023.

Vestues, K. and Rolland, K. (2021). Platformizing the organization through decoupling and recoupling: A longitudinal case study of a government agency. *Scandinavian Journal of Information Systems,* 33 (1): 123-154.

Vishnubhotla, S. D., Mendes, E. and Lundberg, L. (2020). Investigating the relationship between personalities and agile team climate of software professionals in a telecom company. *Information and Software Technology*. Vol 126.

Warley, C.C. Williams, T.D. and Lawler, E.E. II (2014). *The Agility Factor: Building Adaptable Organizations for Superior Performance*. San Francisco, CA: Jossey-Bass.

Wiedemann, A., Gewald, H., Wiesche, M. and Krcmar, H. (2019). Implementing the planning process within DevOps teams to achieve continuous innovation. Conference *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

Wiedemann, A., Wiesche, M., Gewald, H., and Krcmar, H. (2020). Understanding how DevOps aligns development and operations: A tripartite model of intra-IT alignment. *European Journal of Information Systems*, *29*(5), 458–473.

Yin, R.K. (2013). *Case Study Research: Design and Methods*. Thousand Oaks, CA: Sage

Zitkiene, R. and Deksnys, M. (2018). Organizational agility conceptual model. *Montenegrin Journal of Economics, 14*(2), 115–129.

APPENDIX 1. RESPONDENTS

Alpha Respondents

Alpha Respondents	Role in Alpha	Statement in text (theme)
1	Division manager	
2	Department manager	
3	Department manager	
4	Department manager	
5	Department manager	
6	Functional manager	T3, T4
7	Functional specialist, team member	
8	Product owner, team	T2
9	Product owner, team	
10	Division manager	
11	Fuctional specialist, team member	
12	Functional manager	
13	Department manager	
14	Functional specialist, team member	T2, T3
15	Division manager	
16	Functional specialist, team member	T1
17	Department manager	T4
18	Functional specialist, team member	T5
19	Department manager	T4

Beta Respondents

Beta Respondents	Role	Statement in text (theme)
1	Coordinator	
2	Product owner, team	
3	Department manager	
4	Product owner, team	
5	Functional specialist	
6	Functional specialist, team member	T6
7	Product owner, team	
8	Functional specialist, team ember	T2
9	Project owner	
10	Functional specialist	
11	Department manager	
12	Agile coach	

APPENDIX 2. CODING: TOWARDS LONG TERM PROJECTS (T2)

Third-order theme	**Towards long term product teams**		
Second order subtheme	Products or domains are permanent	Team dynamics: Long-term and cross-functional	Gradual change from development project teams
First-order examples of utterances	"For example, we will always have payment and transactions, so we will always need clever people in these areas. When you have lasting domains, it is very sensible to have lasting teams" (Alpha 14).	"You must both operate it and maintain it (in the same team). Now, the logic is: 'If you build it, you own it, you run it, you kill it.' Developers need to think about how to develop a solution that the team can operate and maintain. Employees need a more cross-disciplinary approach . . . and to think more in terms of the life course of a solution. You need to avoid shortcuts that accelerate development but increase the cost of operations" (Alpha 8).	"The change from development projects to products did not take place overnight, but more gradually. We recognized that it was more sensible to work with a long-term horizon in the teams in question . . . I don't remember the exact date, but we gradually started to use the term 'product team' rather than 'project team" . . . It took a long time for parts of the organization to recognize this change" (Beta 8).