

Morteza Moalagh, Marius Mikalsen and Babak A. Farshchian (2023): Are Team Autonomy and Flexibility Enough for Agile Transformation? A Review of Transformed Practices in a Public Sector Organization. In: Proceedings of the 21st European Conference on Computer-Supported Cooperative Work: The International Venue on Practice-centered Computing on the Design of Cooperation Technologies - Exploratory Papers, Reports of the European Society for Socially Embedded Technologies.(ISSN: 2510-2591), DOI: 10.48340/ecscw2023_ep08

Copyright 2023 held by Authors, DOI: 10.48340/ecscw2023_ep08

Permission to make digital or hard copies of part or all 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 the full citation on the first page. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, contact the Authors.

Are Team Autonomy and Flexibility Enough for Agile Transformation? A Review of Transformed Practices in a Public Sector Organization

Morteza Moalagh, Marius Mikalsen and Babak A. Farshchian

Norwegian University of Science and Technology (NTNU), Trondheim, Norway

morteza.moalagh@ntnu.no; marius.mikalsen@ntnu.no; babak.farshchian@ntnu.no

Abstract. The public sector is adopting agile transformation to accelerate digitalization, but it often prioritizes internal efficiency over public values. In this ongoing case study of agile transformation in a large public organization in Norway, we apply a practice-theoretic lens to analyze practice changes. By looking at papers published in recent years on this case, The open coding method was used to identify the components of the practices, including competency, meaning, and material. The study revealed that agile transformation led to a shift in software development practices, improved communication, and increased authority, which enhanced ownership, productivity, and organizational learning. Moreover, The role of architects changed to advisors, and teams took full decision-making authority for managing data and architecture. The analysis of transformed practices revealed that the organization made changes to allow development teams to be more autonomous and flexible in their projects. However, the organization's inherent bias towards development teams seems to hamper effective collaboration and undermines democratic participation, a fundamental value of the public sector. Also, the study illustrates the need for boundary work and infrastructures that integrate the users/business side of the organization beyond the development side to address complex socio-technical interdependencies.

Introduction

Agile software engineering is commonly understood as the application of iterative methods in software development as opposed to planning-based methods such as waterfall models (Sommerville, 2016). Due to their increased popularity, agile methods that originated from software engineering are now being applied to other domains such as teaching (Lang, 2017), project management (Augustine et al., 2005), and organizational change (Fuchs & Hess, 2018). This means that many practices that CSCW researchers are interested in are going through a rapid “agile transformation.” Therefore, CSCW researchers must develop a practice-centered understanding of this phenomenon.

Agile methods often lead to increased autonomy and flexibility at the team level while increasing organizational demands for agility and efficiency. This makes the study of these methods relevant for the CSCW community as they produce tensions between teams and organizations (Tendedez et al., 2018). At the same time, we observe that agile transformation literature frequently ignores the practices of those involved in such transformations, whether they are IT developers, employees, or customers and users.

One area where such agile transformation processes are becoming prevalent is the digitalization of public services. There is increasing pressure on public services due to demographic developments in Western societies and a demand to reduce public spending in many countries. Therefore, agile transformation in the public sector has become a mantra among public administration researchers and practitioners (Mergel et al., 2018).

However, adopting agile methods to transform public services can be a double-edged sword. Most agile methods were initially developed by private companies, whose main outcome was efficiency (Sutherland, 2014). E-government research shows a similar interest in internal organizational efficiency at the expense of other values, such as social and democratic (Twizeyimana & Andersson, 2019). In adopting agile methods to the public sector, it is therefore essential to go beyond

internally oriented imperatives and pay attention to the particular characteristics of the public sector. Public services often exist within complex ecosystems, including citizens in various roles such as recipients of services, providers of services, employees in the public sector, politicians, etc. Public administration, therefore, needs to go beyond internal efficiency as the main value and consider the public value in a broader context (Moore, 1995).

This paper reports an ongoing case study of agile transformation in a large public organization in Norway. The Norwegian Labor and Welfare Administration (abbreviated as NAV) is the Norwegian public welfare agency responsible for providing services like unemployment benefits, pensions, child benefits, and more (NAV, 2022). The case organization has had agile transformation as its main strategy since 2015. Digitalization processes are used to follow a planning-based packaged software acquisition strategy. Now, the organization has set up several autonomous product teams in charge of maintaining digital product families that implement various public services related to, e.g., unemployment, sick leave, and parental leave.

This recent move to agile methods has shown positive results –the case organization has achieved a high level of digitalization and has won several prizes for its innovative services. At the same time, the organization has been criticized for its overemphasis on internal organization and its lack of sensitivity toward the needs of the citizens. A recent report for the Norwegian Board of Health Supervision (Helsetilsynet) uncovered a range of challenges meeting citizens who have to deal with the organization –e.g., lack of access to physical offices, long waiting lists in telephone lines, and lack of communication channels for those with a low level of digital competences (Helsetilsynet, 2020). Moreover, the organization’s latest large-scale digitalization project has met several challenges, such as considerable overspending, delays, and prolonged waiting time for citizens.

Our research interest lies in the intersection of practice-centered studies and agile transformation. We apply a practice-theoretic lens (Shove et al., 2012) to agile transformation in public organizations. Through applying the practice lens, we ask

the question: How do agile transformation practices emerge in public organizations, and how do they affect the provision of public value beyond efficiency? Our research addresses a gap in knowledge by observing and analyzing practice changes.

We build on earlier studies of agile practices (Johannessen & Ellingsen, 2009; Tendedez et al., 2018; Bjørn et al., 2019); and extend these through our study of a large public sector organization and how its practices have changed as the results of its extensive agile transformation initiative. Our preliminary findings are based on a meta-analysis of secondary data from 16 published papers about the case organization. We find that practice changes have led to increased team autonomy and flexibility. However, it is not clear how these changes have impacted the organization's ability to deliver public value.

In the rest of this paper, we first provide an overview of relevant earlier research. Then, we present our findings and discuss them in the light of practice and public value.

Background

Agile Transformation

The evolution of work techniques away from the conventional, linear waterfall development model is agile software development. Agile teams produce work in manageable, tiny batches as opposed to relying solely on a "big bang" release. As a result, there is a shorter time to market, quicker customer requirement adaption, and less risk(S. Rubin, 2013). The agile movement in the software development industry began in 2001 with the agile manifesto's publication by several well-known software practitioners and consultants. Four values were introduced: 1) individuals and interactions over processes and tools, 2) working software over comprehensive documentation, 3) customer collaboration over contract negotiation, and finally, responding to change over following a plan (Abrahamsson

et al., 2002). Every software development agile method or framework, like XP, Scrum, and Kanban, is built upon these values with different degrees of flexibility, speed, learning, and responsiveness (Qumer & Henderson-Sellers, 2006). The idea of self-organizing teams, whose members are collocated and operate at a speed that supports their creativity and productivity, lies at the heart of these methods. The principles support approaches that adapt to changing requirements at any step of the development cycle. Customers are deeply engaged in the creation process, providing feedback and reflection that can produce more satisfactory results (Dingsøyr et al., 2012).

Many organizations thought they could replicate their team-level success at the organizational level after using agile methods, particularly in software firms. The idea of agile transformation was brought up here. Agile transformations were initially created for collocated development teams but are now utilized in various settings (Dingsøyr et al., 2019). This approach encompasses more than just software development techniques and technology; it also refers to new ways of thinking and finding immediate solutions to problems (Olteanu, 2018). In the public sector, agile transformation is used as a technique that can improve the efficiency of service digitalization (Mergel & Ganapati, 2020). According to some studies, implementing agile could enhance public sector collaboration, flexibility, customer satisfaction, and operational excellence (Ylinen, 2021).

Autonomy and Flexibility in CSCW

A key concern in computer-supported collaborative work (CSCW) is how people work together to get things done. Since the early days of workplace studies, studies have demonstrated how introducing digital tools in collaborative work can be a form of *standardization* (Suchman, 1983) and that such systems must be aligned with and support collaborative practices. Failure to do so results in workarounds (Bowers et al., 1995), where workers create practices to bypass the system or engage with them in unexpected and often inefficient ways. This has caused an emphasis on a greater degree of flexibility, where the digital system serves as a tool

for coordinating work while also not overly limiting action. (Blomberg & Karasti, 2013). CSCW, from its origins, is about the tension between standardization and flexibility (Schmidt & Bannon, 2013).

Recently, CSCW research has focused on the tension between standardization and flexibility in software development (Tendedez et al., 2018). Research has shown the importance of coordination, using representations and tools, and organizational obstacles and constraints in software development (Procter et al., 2011). While agile methods have shown success in individual teams and smaller projects, scaling agile methods (such as larger projects or expanding beyond the development unit, i.e., large-scale agile transformation) is known to be challenging (Kruchten, 2013; Hoda, 2019). Scaling agile beyond teams is more challenging because it touches upon or transforms more and different dependencies (i.e., the need to work together to get things done), which traditionally have been controlled through standardization, such as plans (e.g., release plans), documents (e.g., requirement specifications and architecture models), hierarchies (e.g., architecture boards) and formal handovers (e.g., between requirement teams, development teams, and test teams) (Barlow et al., 2011). Example dependencies are between agile teams and from agile teams to the rest of the organization (such as the user side) (Mikalsen et al., 2018). As agile transformation scales, it may be problematic to scale the more flexible agile coordination and communication practices from the team level. Such a revolution may require a broader change in organizational structure and processes (Khan et al., 2016). For example, consider how it can be challenging if a development team is granted more flexibility without the user side of the organization being granted the same flexibility in terms of engaging and participating with the development side. Such one-sided transformations, where the development side is flexible, and the user side is standardized, can harm proper collaboration. It is necessary to consider how the balancing act between standardization and flexibility plays out in agile transformations. In so doing, we question existing conceptions of agile transformation and point to a suitable challenge for CSCW.

Practice Theory

Given that agile transformation is a cultural shift closely related to cooperation and communication between various organizational stakeholders, one appropriate method to investigate the effects of this transformation is to examine organizational practices and procedures (Naslund & Kale, 2020). Practice theory examines how practices are carried out, sustained, and changed in various organizational and social settings (Schmidt, 2018). A practice lens is necessary to understand the routines better, activities, and tasks that people perform while working collaboratively, as well as how these practices are influenced by the tools and technologies they employ (Bourdieu, P, 1997; Schatzki et al., 2005).

As we will see later, we have used practice theory to identify and analyze a set of practices in published literature. To understand the practices of agile transformations better, we look to cultural theories of practice. Cultural theories of practice "highlight the significance of shared or collective symbolic structures of knowledge in order to grasp both action and social order" (Reckwitz, 2002, p.246). Shove et al. (2012), based on Reckwitz (2002) cultural orientation, defined *meaning*, *material*, and *competency* as the components of practices. According to Shove et al. (2012), meaning refers to socially accepted theories or notions about a practice that gives it purpose, justifications for doing it, and rationalizations for why. Competency includes knowledge as well as attitudes and behaviors. It also provides insights into what is right, normal, and acceptable. The third element, material, refers to the physical things, innovations, and built environments that people use on a daily basis to enact their practices (Table I). Integrated components make up practices. As connections between their distinguishing elements are made and broken, practices transform, endure, and eventually vanish. Wherever these components are connected, they will be incorporated into one another, and practices will be developed. As Shove et al. put it, each component, or "element," shapes the others. When practices change, the component elements of each practice are either scrapped completely, get it passive and shrouded, or altered to become a part of the new practice or other new practices (Shove et al., 2012).

Table I- Definitions of practice components (Shove et al., 2012)

Practice Component	Definition
Competency	It includes knowledge as well as attitudes and behaviors. Competency also provides insights into what is right, normal, and acceptable.
Meaning	It refers to socially accepted theories or notions about a practice that gives it purpose, justifications for doing it, and rationalizations for why.
Material	It refers to the physical things, innovations, and built environments that people use on a daily basis to enact their practices.

Case Description

NAV manages a third of the Norwegian budget and employs over 19000 people to provide a portfolio of hundreds of services. The administration offers welfare services to the Norwegian population, such as job security, unemployment benefit, family-related benefit, occupational injury, health services, financial advice and debt counseling, and retirement and disability benefit (NAV, 2022). As established in 2006, NAV is a complex organization, or onion, as some scholars refer to it, with multiple layers of line organization structure. Its executives hoped to modernize its services through the use of new technologies. Between 2012 and 2016, several factors, such as the Norwegian government's strategy for digitizing citizen services, the failure of some projects that were put into action, and unfavorable media coverage of the company's performance in creating systems that matched users' needs led the organization's senior managers to hold specialized meetings and come to a conclusion that the organization's strategy in this area needed to be changed (Kohansal & Haki, 2021). By employing 800 people in the IT department, the company could digitalize many services by shifting its strategy to insourcing application development and modernizing architecture. NAV has created a platform that allows different teams to automatically deploy their applications. Agile methodologies began to be implemented at the team level by 2016 due to issues such as a long time to market, low user satisfaction, and a lack of internal understanding of the procedures and systems. In order to organically create new solutions, the company changed its IT ecosystem in 2017. This was a significant

move that impacted the organization, tools, people, and technologies employed by the NAV IT department (Mikalef & Parmiggiani, 2022).

NAV initially consisted of small, agile teams comprising product owners and development team members. The business decided to run fewer projects and concentrate more on improving current products with the help of agile mentors. They can choose their strategy by creating autonomous teams. With approximately 2.8 million people as users, NAV has developed product areas that are funded independently. Since the company switched teams to product areas at the end of 2019, most IT staff members are now working on specialized product teams or teams that offer infrastructure and cross-border services. The organization now has seven cross-functional product teams (CFPT) made up of 2 to 9 teams, four specialized product areas made up of 5 to 14 teams each, and 15 CFPTs that are awaiting status determination. This transition from an antiquated IT department to a product-focused company has been compared to a toddler growing into an adult by some senior executives. Due to the aforementioned modifications, the weekly release volume has increased significantly from 50 to 800, the roles of functional advisers and developer and designer have been switched, and internal IT staff members now feel more ownership of their work (Mohagheghi & Lassenius, 2021).

Method

This paper reports preliminary results from an ongoing case study (Yin, 2013). The phenomenon we study is the changes in practices as a consequence of agile transformation and their potential impact on public value. Our study is interpretative (Walsham, 1995) as we are currently trying to interpret our data in the context of several plausible theories in parallel. Our current status can be called a framing phase, where we are trying to conceptualize the phenomenon and collect and organize initial data (Pan & Tan, 2011).

The findings reported here are based on a meta-analysis of secondary data from research that various researchers have published about agile transformation in

NAV. We are aware that the use of second-hand data may be interpreted in such a way that the article is a systematic literature review. However, this article is presented as an exploratory paper and is part of ongoing research with a case study strategy that will use first-hand data such as observations and interviews in future full versions. Some of the publications used here are by researchers in our network. Others we have obtained by doing systematic searches in various databases (See Table II). After screening, we were left with 16 published papers that constitute the basis for our findings in this paper. We included only qualitative papers that discussed agile transformation in NAV. These articles were mainly written from 2016 onwards, and each described the agile transformation in the NAV from a different point of view.

Table II- Systematic Search Results in Different Scientific Databases

Keyword	Web of Science	Scopus	IEEE Explorer	Science Direct	ProQuest	AIS
TITLE-ABS-KEY (Norwegian OR welfare OR agile)	7	2	1	227	393	15
TITLE-ABS-KEY (NAV OR agile)	67	10	2	123	364	12

We analyzed the papers based on the model of practice from Shove et al. (2012). To do this, we first identified and coded all the practices studied in our sample. We then analyzed these practices to find the competencies, meaning, and material each practice contained and, where possible, the changes in these components. We used open coding. Words and sentences that discussed practices and their component units were extracted, grouped, and then used with quotations to describe the practices, their status, and the constituent parts of each practice. The results are shown in Tables III to V in the Findings section.

We acknowledge the limitations inherent in secondary data analysis, as we base our findings on limited data without ourselves having been exposed to the cases reported in these papers (Small & McCrory Calarco, 2022). We need to rely on the interpretations of other researchers instead of our own. On the other hand, secondary research-based sources as the first source of data for our case study can

also provide advantages. Analysis of secondary data, as we have done here, provides a broad perspective of NAV's agile transformation beyond what is available in internal policy and strategy documents and media coverage. As a first data source, they provide valuable insight into NAV and how other researchers have perceived NAV's agile transformation. We plan to add new data in the form of interviews with NAV employees, additional document analysis –both internal but also external media coverage – and observations of product teams.

Findings

In this section, as seen in Tables III to V, the open coding method was used to identify all of the identified practices' components, including competence, meaning, and material. For each component in these tables, pertinent quotations from the papers that have been reviewed serve as supporting evidence.

Software Development Practices

Following its gradual introduction in the NAV organization in 2016, the agile approach slowly transformed and matured, leading to significant changes in many operational aspects. One of these crucial changes was the shift of the software development practices from a project approach to a product-oriented approach (Table III). The shift from project-focused to product-focused improved communication between the business and information technology departments (Mikalef & Parmiggiani, 2022). From the lens of practice theory, through the competency component, Prior to the agile transformation, the information technology department for software development served primarily as a middleman and project supervisor, but after internalizing production and undergoing the scaled-agile transformation, product areas were gradually established, and specialized teams were formed around each product (Kohansal & Haki, 2021; Mohagheghi & Lassenius, 2021). Before the transformation, The information technology department collected the requirements from different business departments and,

after reviewing and summarizing, sent them to the system development contractors (Dingsøyr et al., 2022). The increase in the authority granted to groups of people in the form of solution development teams was another significant change that took place within the software development practice in NAV. The advantages of increasing authority include lowering organizational stress, raising motivation, enhancing interpersonal relationships, and raising productivity (Stray et al., 2018).

From the lens of practice theory, the meaning of software development in NAV has also undergone a significant change. Before the transformation, every project was carried out using a special methodology established in 2012 and involved numerous steps and handoffs between the IT department, business units, and vendors (Mohagheghi & Lassenius, 2021). The development scope was fixed, and the releases were too large and consisted of too many features (Lundene & Mohagheghi, 2018a). The sense of ownership of the products increased due to the organizational approach being changed to become more product-oriented. Product development proceeded more quickly, and organizational resources were not wasted (Kohansal & Haki, 2021). The NAV organization adopted an agile approach that increased organizational learning and reduced the gap between the business and information technology departments, allowing these two departments to coordinate their efforts and form specialized teams to focus on product development (Lundene & Mohagheghi, 2018a; Dingsøyr et al., 2023).

From the material viewpoint, The NAV software development infrastructure underwent a complete transformation from a material standpoint. Before the transformation, there were more than 50 systems in the company that was created by various contractors using numerous user interfaces and programming languages (Paasivaara & Kruchten, 2020). A product manager was assigned to each project who was responsible for creating the product backlog, gathering the requirements identified during meetings with business users, and managing them using the Jira software (Lundene & Mohagheghi, 2018a). Following the transformation, NAV concentrated on internal infrastructure development to create information systems and then established the NAIS platform. This platform allowed different

software development teams to continuously release software (Mikalef & Parmiggiani, 2022). Other platforms developed by NAV include Night Watch allows teams to independently design, execute, and implement software tests (Lundene & Mohagheghi, 2018a). The organization's goal in designing these platforms is to establish an environment for continuous software production and getting user feedback (Dingsøyr et al., 2023).

Table III – Software Development Practices and their components in NAV

Practice Component	Quotes From Literature
Old Practice: Project-Oriented Development	
Competency	<ul style="list-style-type: none"> • “The requirements were described through epics, which were broken down into user stories and detailed by domain experts. The user stories were then handed over to the development teams”(Dingsoyr et al., 2022, p.2) • “The IT department worked as a middleman between the business departments, from which funding and requirements came, and the vendors who did the actual development. The main roles in the IT department assisting system owners in contact with vendors were functional advisors, who had business domain knowledge and assisted in tasks such as analysis and estimation, and technical advisors”(Mohagheghi & Lassenius, 2021, p.1447)
Meaning	<ul style="list-style-type: none"> • “Up till 2016, nearly all software development and maintenance were conducted as large coordinated releases. These were typically organized in a staged workflow with defined phases and hand-overs. The teams in this process had a limited degree of autonomy: The scope of the iteration was fixed, the teams had to follow common processes, and they delivered to large planned releases and had to coordinate with other teams through the release plans.” (Lundene & Mohagheghi, 2018a, p.2) • “The development process used in all development was defined in 2012 and was a waterfall process with many gates and handovers between the client and vendor” (Mohagheghi & Lassenius, 2021, p.1448)
Material	<ul style="list-style-type: none"> • “Over 50 applications covering a broad range of user interfaces were managed by a group of employees organized in an office in the IT department. The office managed the contract with the vendor, provided support to the business side, and followed testing, deployment, and operations of the applications.” (Paasivaara & Kruchten, 2020, p.247) • “Typically, A project was started for new development or major modifications, and the user stories assigned to the project were organized

Practice Component	Quotes From Literature
	by product owners in a prioritized queue with virtual backlogs for each team using the JIRA software” (Lundene & Mohagheghi, 2018a, p.2)
New Practice: Product-Oriented Development	
Competency	<ul style="list-style-type: none"> • “A new organizational model, known as “Product Area,” was introduced to NAV. The Product Area groups a number of cross-functional teams and is in charge of one or more "user journeys" or "user groups," as well as related products or features. A Product Area encompasses all of the skills required to manage, develop, deliver, and maintain software within its responsibility scope.” (Kohansal & Haki, 2021, p.6) • “Placing both business and development people in cross-functional teams led to fewer handovers, and requirement dependencies, in particular, were managed at a low level.” (Dingsøy et al., 2023, p.39)
Meaning	<ul style="list-style-type: none"> • “From 2017 to 2020, NAV has moved from project development to ongoing product development in interdisciplinary/cross-functional product teams.” (Mikalef & Parmiggiani, 2022, p.128) • “Silo systems are decoupled into modular applications, and the hierarchical organization recoupled into cross-functional teams that perform continuous software development.” (Vestues & Knut, 2019, p.6) • “Teams have taken the responsibility for deployment as well, and most teams release software continuously.” (Mohagheghi & Lassenius, 2021, p.1451)
Material	<ul style="list-style-type: none"> • “By developing its own application platform for deployment/production setting (NAIS), NAV IT gained a modern application platform that allowed teams to automatically deploy to production whenever they wanted.” (Mikalef & Parmiggiani, 2022, p.127) • “The teams use Night Watch, an end-to-end testing tool, to develop and run automated functional tests. As an autonomous cross-functional team, all team members owned and were responsible for the product quality” (Lundene & Mohagheghi, 2018b, p.4) • “A new technical platform enabled continuous delivery, which increased the feedback speed.” (Dingsøy et al., 2023, p.39)

Enterprise Architecture Practices

Another practice that underwent significant modifications due to the agile transformation in the NAV organization was enterprise architecture(EA). As Table IV shows, architects played important roles at all stages before adopting agile

frameworks. From the lens of competency, they were management assignees and so-called technical police who monitored everything, and vendors should get their approval before any roll-out. Architects had significant executive power and controlled the projects' budget, timeline, and execution strategy. Enterprise architects, notwithstanding, refuted this assertion and proclaimed that although architects may have assumed the manager's position, it is likely that they did so under the manager's oversight (Kohansal & Haki, 2021). At that time, the administration established a specific section for EA to guarantee that all projects adhere to architectural procedures and practices. This section created and disseminated governance frameworks and architectural decisions, such as integration architecture or security standards, which all projects must follow (Sortehaug Ajer & Olsen, 2018). However, after the agile transformation, the role of architects changed to advisors, and the teams took full decision-making authority on the choice of architecture. Some architects joined agile product development teams, reducing the size of the central team and shifting its role from supervisory to advisory (Kohansal & Haki, 2021).

The enterprise architecture's meaning component significantly changed after the agile transformation. Previously, architecture focused more on project budget, cost, and timeline control, as well as ensuring that each project adhered to the organization's standards. EA, however, has shifted its attention more toward product areas since going agile. Teams choose the development frameworks and infrastructures to be used; there is no longer a predetermined architectural framework for creating systems (Kohansal & Haki, 2021). From the lens of material components, EA practices have evolved from controlling integration to modular architecture (Vestues & Knut, 2019) to give teams more freedom in creating their applications.

Table IV – Enterprise Architecture Practices and their Components in NAV

Practice Component	Quotes From Literature
Old Practice: Top-Down Enterprise Architecture	
Competency	<ul style="list-style-type: none"> • “In the NAV, external EA consultants had made plans not aligned with the NAV business. However, an architect from the NAV explained that the management had realized that ‘we need our own employees to be the ones who ensure continuity and, for example, ensure proper documentation.’” (Ajer & Olsen, 2019, p.88) • “The NAV has established an EA section to align the programs and the projects with the NAV’s long-term goals. NAV have established separate functions for architecture and design that work in an interdisciplinary manner across programs and projects to help in complicated situations and ensure that architectural practices are followed.” (Sortehaug Ajer & Olsen, 2018, p.5)
Meaning	<ul style="list-style-type: none"> • “The departments worked independently in the NAV, without much interaction. An enterprise architect noted, ‘NAV is a strong line-driven organization, [with] very little matrix focus. A [horizontal perspective] has very little authority and power in practice, and the hierarchy in government organizations reinforces this.’” (Sortehaug Ajer & Olsen, 2018, p.7) • “The IT projects were primarily governed by function, costs, and deadlines, necessitating close monitoring of these factors. In this governance model, architects, at all levels, had strong roles. Some believed that architects had taken the manager's role. Thus, the command and control were their approaches, and their role was assumed as the technology police, which the vendor companies only tried to get their approval.” (Kohansal & Haki, 2021, p.6)
Material	<ul style="list-style-type: none"> • “The IT department was responsible for a number of tasks, including the creation of high-level constraints such as integration architecture and security specifications, contract management, operational and technical assistance to business units, owning the system for integration and release, providing first customer support, and ensuring the service's full functionality.” (Kohansal & Haki, 2021, p.5)
New Practice: Federated (Decentralized) Enterprise Architecture	
Competency	<ul style="list-style-type: none"> • “Architects' contribution to the agile teams by bringing holistic view and organizational knowledge.” (Kohansal & Haki, 2021, p.12) • “As a result, a new experience of collaboration between architects and teams was shaped by the collaboration of architects within the agile teams. Thus, agile teams recognized that the architect's experiences and skills could be a required resource. However, the architects chose not to use the

Practice Component	Quotes From Literature
	title "architect" in their collaboration. They anticipated that this title would create some difficulties. Through this collaboration, several architects confirm that they previously lacked sufficient flexibility in their jobs. They also agreed that in order to create value in this new way of working, they must reform their working style.” (Kohansal & Haki, 2021, p.11)
Meaning	<ul style="list-style-type: none"> • “Agile teams are more accepting of the importance of EA practices. 40 designers were hired to complement the necessary team skills, and decentralized architectural work was introduced. Enterprise architects also proposed a new organizational structure called Product Area at this time. It was an effective way of establishing agile teams that was well-received by individuals.” (Kohansal & Haki, 2021, p.10)
Material	<ul style="list-style-type: none"> • “The modular architecture of the application platform was a precondition for the ongoing change. By decoupling applications, development teams could develop and release applications frequently, ensuring continuous feedback from systems and users.” (Vestues & Knut, 2019, p.10)

Data Management Practices

Data management practices are the next domain that has changed dramatically after the agile transformation. As shown in Table V, From the competency perspective, the knowledge management department was responsible for creating data storage models and analytical reports before internalization and agile transformation. However, with the swift development of software, the rapid modification of data models, and the fact that changes are unknown, the responsibility has become decentralized. Each team was in charge of managing the data. In the past, data management practices revolved around gathering data from various systems and transforming it into a data warehouse. However, the concept of "data product"—which refers to managing data by users' needs and creating a strategy for its upkeep and management—emerged as a result of agile transformation. From the lens of material, the business also created a platform that teams could use on their own to share data and create the knowledge products the business required, like dashboards (Vestues et al., 2022). The ongoing updating of data necessitates the replacement of outdated plans, which is impossible with centralized data management and

maintenance models. Traditional data management models, which gather analytical data in data silos and analyse it centrally, are incompatible with the ongoing deployment approach of agile software development (Stray et al., 2022).

Table V – Data Management Practices and their components in NAV

Practice Component	Quotes From Literature
Old Practice: Centralized Data Management	
Competency	<ul style="list-style-type: none"> “Within NAV, Analytical data has traditionally been managed by a single unit, the Knowledge department. As the name implies, the Knowledge department has been responsible for producing analytical insight about NAV, ranging from public statistics to internal steering information.” (Stray et al., 2022, p.226)
Meaning	<ul style="list-style-type: none"> “The Knowledge department has traditionally been responsible for gathering analytical data across NAV. These data have been extracted from source systems, transformed, and loaded into a data warehouse.” (Stray et al., 2022, p.226)
Material	<ul style="list-style-type: none"> “The data warehouse team has been responsible for transforming and compiling data into a coherent data model. This requires extensive knowledge of both source systems and business domains” (Stray et al., 2022, p.226)
New Practice: Distributed Data Management	
Competency	<ul style="list-style-type: none"> “For NAV, a shift from centralized to decentralized ownership implies that application development teams assume responsibility for their own data. With the distributed data ownership, interpretations and decisions relating to the data are done by the people closest to the data.” (Stray et al., 2022, p.227)
Meaning	<ul style="list-style-type: none"> “As a means of implementing data ownership, teams will develop so called “data products.” A data product is defined as a dataset and the documentation it. Data products require deliberate design and management, satisfying the needs of prospective users” (Stray et al., 2022, p.227)
Material	<ul style="list-style-type: none"> “By offering the data as reusable services, the data could then be reused by other applications and services – thereby increasing the accuracy and efficiency of transactions.” (Vestues & Knut, 2019, p.11) “To enable distributed data ownership, the organization has introduced a self-serve data platform called NADA. While data in the data warehouse is collected and curated by a single team, the new data platform offers functionality which allows all teams to share their data. The NADA platform is thus a multisided platform where the entire organization can produce and consume data” (Stray et al., 2022, p.228)

Discussion

Agile Transformation has become the belle of the ball in project and product management techniques in the past decade, gaining immense popularity in organizations worldwide. It has been the go-to approach for many, and now, the public sector is also trying to jump on the bandwagon. The goal is to accelerate and enhance the digitalization of their services, hoping to catch up with the rest of the world. However, the central query here is how agile methodologies can be applied in public organizations and what impact, beyond efficiency, they have on public values. As a cultural shift that inevitably impacts how people interact, communicate, and collaborate within an organization, it requires a deep dive into practices to understand its effects fully. Our study is about understanding how the agile approach shapes public values.

This article documents our initial analysis of the changes made within the NAV organization, using a case study approach to reveal critical findings. Through an investigation of various practices outlined in the findings, it becomes clear that software development has undergone a significant transformation. Internal agile teams have emerged, with the increased authority to design, develop, and maintain their products. In turn, platforms have been established to facilitate continuous software development, allowing teams to adopt the most suitable architectural options. As a result, the role of the organizational architecture department has shifted from hands-on development to that of a consultant and advisor. Service-oriented and modular architectures have been employed to further empower the development teams by providing more freedom and control over creating digital services. This autonomy extends to knowledge mining, business intelligence development, and data management. The development teams completely own data products, from designing and implementing reporting structures to creating analytical frameworks. All of these changes serve to enhance the agility of the organization, ultimately leading to more efficient and effective software development.

The analysis of NAV-transformed practices revealed two key elements: *authority* and *flexibility*. The organization changed to allow development teams to be more autonomous and flexible in their projects. However, the organization's inherent bias towards development teams and internal efficiency seems to hamper effective collaboration and undermine democratic participation, a fundamental value of the public sector. According to Saldivar et al. (2019), Democratic participation involves including all stakeholders in the decision-making process and building consensus rather than relying solely on the power and authority of a select few. In the realm of CSCW, a commitment to democratic participation and collaboration is crucial to creating an inclusive and innovative environment that fosters sustained growth and progress. With the expansion of agile approaches to scale in the organization and the creation of different teams for multiple product areas, this gap will gradually deepen. Introducing flexibility into a vast and intricate organization that has grown accustomed to relying on documents, plans, and hierarchies can pose a significant challenge to the organization's ability to regulate and oversee development. This conundrum can also be quite bothersome for development teams (Barlow et al., 2011; Hanseth et al., 1996).

The review of papers found that the NAV agile transformation is an internal/administrative development. The organization is undergoing a large-scale agile transformation, which poses significant challenges to conventional notions of large-scale agile implementation. This transformation requires the development of boundary and address the “*boundary work and boundary infrastructures that are required for working across contexts resolving and coordinating complex socio-technical interdependencies*” (Rolland et al., 2016, p.2). This discussion points to how we must expand notions of large-scale agile transformation to move beyond internal/admin concerns like internal efficiency and address public value. A key issue here is to broaden from a focus on only the dev side and a sole focus on autonomy and flexibility and embrace more complexity to address the boundary work involving the users/business side of the organization.

Conclusion

In pursuing digital transformation, public organizations are increasingly turning to agile methodologies to enhance their efficiency and catch up with the rest of the world. While agile transformation has proven successful in many organizations, applying it to the public sector requires a deeper understanding of its impact on public values, including democratic participation and collaboration. Through a case study approach, we revealed the significant transformation of software development within the NAV organization. The emergence of internal agile teams with increased authority over the choice of architecture, design, development, and maintain complex systems enhanced the agility of the organization, leading to more efficient and effective software development. This approach is a critical factor for public organizations to provide excellent services.

The case study of NAV reveals that although agile transformation has enabled development teams to become more autonomous and flexible, it has also highlighted inherent biases towards development teams, which may undermine democratic participation. To address this issue, there is a need to broaden the focus of agile transformation beyond internal/administrative concerns. A more nuanced and complex understanding of agile transformation is required, one that considers public organizations' unique challenges and opportunities. As we delve deeper into agile transformation, it becomes apparent that the key to success lies in more than just studying and interviewing the development teams. It is imperative that we also engage with employees from other departments within the organization, end users, and even citizens who have utilized the services provided. These valuable insights and perspectives can be integrated into the agile transformation design process. We will undoubtedly explore this further in the subsequent phases of our research.

The NAV case study also illustrates the need for boundary work and infrastructures that integrate the users/business side of the organization beyond the development side to address complex socio-technical interdependencies. Embracing complexity and developing a more holistic approach to agile transformation can facilitate

sustained growth and progress, creating an inclusive and innovative environment that fosters public values and enhances the digitalization of public services. As we look to the future, it seems essential that agile transformation studies shift their emphasis toward closely examining how public institutions change in relation to how they align with public values. This type of research will be of great value to organizations as they work to comply with government regulations regarding the wise distribution of public funds in line with democratic principles.

References

- Abrahamsson, P., Salo, O., & Ronkainen, J. (2002). *Agile Software Development Methods: Review and Analysis*. 112.
- Ajer, A. K. S., & Olsen, D. H. (2019). Enterprise Architecture Implementation Is a Bumpy Ride: A Case Study in the Norwegian Public Sector. *Electronic Journal of E-Government*, 17(2). <https://doi.org/10.34190/EJEG.17.2.002>
- Augustine, S., Payne, B., Sencindiver, F., & Woodcock, S. (2005). *Agile project management*.
- Barlow, J. B., Giboney, J. S., Keith, M. J., Wilson, D. W., Schuetzler, R. M., Lowry, P. B., & Vance, A. (2011). Overview and Guidance on Agile Development in Large Organizations. *Communications of the Association for Information Systems*, 29, 25–44.
- Bjørn, P., Sørderberg, A.-M., & Krishna, S. (2019). Translocality in Global Software Development: The Dark Side of Global Agile. *Human-Computer Interaction*, 34(2), 174–203. <https://doi.org/10.1080/07370024.2017.1398092>
- Blomberg, J., & Karasti, H. (2013). Reflections on 25 Years of Ethnography in CSCW. *Computer Supported Cooperative Work (CSCW)*, 22(4–6), 373–423. <https://doi.org/10.1007/s10606-012-9183-1>
- Bourdieu, P. (1997). *Outline of a Theory of Practice*. Cambridge University Press.
- Bowers, J., Button, G., & Sharrock, W. (1995). Workflow From Within and Without: Technology and Cooperative Work on the Print Industry Shopfloor. In H. Marmolin, Y. Sundblad, & K. Schmidt (Eds.), *Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work ECSCW '95* (pp. 51–66). Springer Netherlands. https://doi.org/10.1007/978-94-011-0349-7_4
- Dingsoeyr, T., Falessi, D., & Power, K. (2019). Agile Development at Scale: The Next Frontier. *IEEE Software*, 36(2), 30–38. <https://doi.org/10.1109/MS.2018.2884884>
- Dingsøy, T., Bjørnson, F. O., Schrof, J., & Sporsen, T. (2023). A longitudinal explanatory case study of coordination in a very large development programme: The impact of

- transitioning from a first- to a second-generation large-scale agile development method. *Empirical Software Engineering*, 28(1), 1. <https://doi.org/10.1007/s10664-022-10230-6>
- Dingsøyr, T., Jorgensen, M., Carlsen, F. O., Carlstrom, L., Engelsrud, J., Hansvold, K., Heibo-Bagheri, M., Roe, K., & Sorensen, K. O. V. (2022). Enabling Autonomous Teams and Continuous Deployment at Scale. *IT Professional*, 24(6), 47–53. <https://doi.org/10.1109/MITP.2022.3209871>
- Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213–1221. <https://doi.org/10.1016/j.jss.2012.02.033>
- Fuchs, C., & Hess, T. (2018). *Becoming Agile in the Digital Transformation: The Process of a Large-Scale Agile Transformation*. 18.
- Hanseth, O., Monteiro, E., & Hatling, M. (1996). Developing Information Infrastructure: The Tension Between Standardization and Flexibility. *Science, Technology, & Human Values*, 21(4), 407–426. <https://doi.org/10.1177/016224399602100402>
- Helsetilsynet. (2020). Veileder for landsomfattende undersøkelse 2020-2021 om tilgjengelighet til sosiale tjenester i Nav. Internserien 2/2020.
- Hoda, R. (Ed.). (2019). *Agile Processes in Software Engineering and Extreme Programming – Workshops: XP 2019 Workshops, Montréal, QC, Canada, May 21–25, 2019, Proceedings (Vol. 364)*. Springer International Publishing. <https://doi.org/10.1007/978-3-030-30126-2>
- Johannessen, L. K., & Ellingsen, G. (2009). Integration and Generification—Agile Software Development in the Healthcare Market. *Computer Supported Cooperative Work (CSCW)*, 18(5–6), 607–634. <https://doi.org/10.1007/s10606-009-9097-8>
- Khan, M. R., Fernández, W. D., & Jiang, J. J. (2016). Is there such a thing as agile IT program management?
- Kohansal, M. A., & Haki, K. (2021). Enterprise Architecture’s Ups and Downs Over Time: A Case of De- and Re- Institutionalization.
- Kruchten, P. (2013). Contextualizing agile software development: CONTEXTUALIZING AGILE SOFTWARE DEVELOPMENT. *Journal of Software: Evolution and Process*, 25(4), 351–361. <https://doi.org/10.1002/smr.572>
- Lang, G. (2017). Agile Learning: Sprinting Through the Semester. *Information Systems Education Journal*, 15(3), 14–21.
- Lundene, K., & Mohagheghi, P. (2018a). How autonomy emerges as agile cross-functional teams mature. *Proceedings of the 19th International Conference on Agile Software Development: Companion*, 1–5. <https://doi.org/10.1145/3234152.3234184>
- Lundene, K., & Mohagheghi, P. (2018b). How autonomy emerges as agile cross-functional teams mature. *Proceedings of the 19th International Conference on Agile Software Development: Companion*, 1–5. <https://doi.org/10.1145/3234152.3234184>

- Mergel, I., & Ganapati, S. (2020). Agile: A New Way of Governing. *Public Administration Review*, 81(1), 161–165.
- Mergel, I., Gong, Y., & Bertot, J. (2018). Agile government: Systematic literature review and future research. *Government Information Quarterly*, 35(2), 291–298. <https://doi.org/10.1016/j.giq.2018.04.003>
- Mikalef, P., & Parmiggiani, E. (Eds.). (2022). *Digital Transformation in Norwegian Enterprises*. Springer International Publishing. <https://doi.org/10.1007/978-3-031-05276-7>
- Mikalsen, M., Moe, N. B., Stray, V., & Nyrud, H. (2018). Agile Digital Transformation: A Case Study of Interdependencies. 9.
- Mohagheghi, P., & Lassenius, C. (2021). Organizational implications of agile adoption: A case study from the public sector. *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 1444–1454. <https://doi.org/10.1145/3468264.3473937>
- Moore, M. H. (1995). *Creating public value: Strategic management in government*. Harvard university press.
- Naslund, D., & Kale, R. (2020). Is agile the latest management fad? A review of success factors of agile transformations. *International Journal of Quality and Service Sciences*, 12(4), 489–504. <https://doi.org/10.1108/IJQSS-12-2019-0142>
- NAV. (2022). What is NAV? <https://www.nav.no/en/home/about-nav/what-is-nav>
- Olteanu, C. G. (2018). IT Agile Transformation. *Academy of Economic Studies. Economy Informatics*, 18(1), 23–31.
- Paasivaara, M., & Kruchten, P. (Eds.). (2020). *Agile Processes in Software Engineering and Extreme Programming – Workshops: XP 2020 Workshops*, Copenhagen, Denmark, June 8–12, 2020, Revised Selected Papers (Vol. 396). Springer International Publishing. <https://doi.org/10.1007/978-3-030-58858-8>
- Pan, S. L., & Tan, B. (2011). Demystifying case research: A structured–pragmatic–situational (SPS) approach to conducting case studies. *Information and Organization*, 21(3), 161–176. <https://doi.org/10.1016/j.infoandorg.2011.07.001>
- Procter, R., Rouncefield, M., Poschen, M., Lin, Y., & Voss, A. (2011). Agile Project Management: A Case Study of a Virtual Research Environment Development Project. *Computer Supported Cooperative Work (CSCW)*, 20(3), 197–225. <https://doi.org/10.1007/s10606-011-9137-z>
- Qumer, A., & Henderson-Sellers, B. (2006). COMPARATIVE EVALUATION OF XP AND SCRUM USING THE 4D ANALYTICAL TOOL (4-DAT). 9.
- Reckwitz, A. (2002). Toward a Theory of Social Practices: A Development in Culturalist Theorizing. *European Journal of Social Theory*, 5(2), 243–263. <https://doi.org/10.1177/13684310222225432>

- Rolland, K. H., Dingsøy, T., & Fitzgerald, B. (2016). *Large-Scale Agile Development*. Thirty Seventh International Conference on Information Systems, Dublin.
- S. Rubin, K. (2013). *ESSENTIAL SCRUMA PRACTICAL GUIDE TO THE MOST POPULAR AGILE PROCESS*. Pearson Education.
- Saldivar, J., Parra, C., Alcaraz, M., Arteta, R., & Cernuzzi, L. (2019). Civic Technology for Social Innovation: A Systematic Literature Review. *Computer Supported Cooperative Work (CSCW)*, 28(1–2), 169–207. <https://doi.org/10.1007/s10606-018-9311-7>
- Schatzki, T. R., Savigny, E. von, & Knorr-Cetina, K. (2005). *The Practice Turn in Contemporary Theory*. Taylor and Francis.
- Schmidt, K. (2018). Practice and Technology: On the Conceptual Foundations of Practice-Centered Computing. In *Socio-Informatics* (pp. 47–104). Oxford University Press.
- Schmidt, K., & Bannon, L. (2013). Constructing CSCW: The First Quarter Century. *Computer Supported Cooperative Work (CSCW)*, 22(4–6), 345–372. <https://doi.org/10.1007/s10606-013-9193-7>
- Shove, E., Pantzar, M., & Watson, M. (2012). *The dynamics of social practice: Everyday life and how it changes*. SAGE.
- Small, M. L., & McCrory Calarco, J. (2022). *Qualitative literacy: A guide to evaluating ethnographic and interview research*. University of California Press.
- Sommerville, I. (2016). *Software engineering* (10. ed., global ed). Pearson.
- Sortehaug Ajer, A. K., & Olsen, D. H. (2018). *ENTERPRISE ARCHITECTURE CHALLENGES: A CASE STUDY OF THREE NORWEGIAN PUBLIC SECTORS*. 17.
- Stray, V., Moe, N. B., & Hoda, R. (2018). Autonomous agile teams: Challenges and future directions for research. *Proceedings of the 19th International Conference on Agile Software Development: Companion*, 1–5. <https://doi.org/10.1145/3234152.3234182>
- Stray, V., Stol, K.-J., Paasivaara, M., & Kruchten, P. (Eds.). (2022). *Agile Processes in Software Engineering and Extreme Programming: 23rd International Conference on Agile Software Development, XP 2022, Copenhagen, Denmark, June 13–17, 2022, Proceedings* (Vol. 445). Springer International Publishing. <https://doi.org/10.1007/978-3-031-08169-9>
- Suchman, L. A. (1983). Office procedure as practical action: Models of work and system design. *ACM Transactions on Information Systems*, 1(4), 320–328. <https://doi.org/10.1145/357442.357445>
- Sutherland, J. V. (2014). *Scrum: The art of doing twice the work in half the time* (First Edition). Crown Business.
- Tendedez, H., Ferrario, M. A. M. A. F., & Whittle, J. (2018). Software Development and CSCW: Standardization and Flexibility in Large-Scale Agile Development. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW), 1–23. <https://doi.org/10.1145/3274440>

- Twizeyimana, J. D., & Andersson, A. (2019). The public value of E-Government – A literature review. *Government Information Quarterly*, 36(2), 167–178. <https://doi.org/10.1016/j.giq.2019.01.001>
- Vestues, K., Hanssen, G. K., Mikalsen, M., & Buan, T. A. (Eds.). (2022). A Case Study. In *Agile Processes in Software Engineering and Extreme Programming*. In *Agile Processes in Software Engineering and Extreme Programming* (Vol. 445, pp. 220–235). Springer International Publishing. <https://doi.org/10.1007/978-3-031-08169-9>
- Vestues, K., & Knut, R. (2019). Making Digital Infrastructures More Generative Through Platformization and Platform- driven Software Development: An Explorative Case Study.
- Walsham, G. (1995). Interpretive case studies in IS research: Nature and method.
- Yin, R. K. (2013). Validity and generalization in future case study evaluations. *Sage Publication*, 19, 321–332.
- Ylinen, M. (2021). Incorporating agile practices in public sector IT management: A nudge toward adaptive governance. *Information Polity*, 26(3), 251–271. <https://doi.org/10.3233/IP-200269>