

How Enterprises Adopt Agile Structures: A Multiple-Case Study

Daniel Gerster
University of St.Gallen,
Switzerland
daniel.gerster@student.unisg.ch

Christian Dremel
University of St.Gallen,
Switzerland
christian.dremel@unisg.ch

Walter Brenner
University of St.Gallen,
Switzerland
walter.brenner@unisg.ch

Prashant Kelker
Information Services
Group, Germany
prashant.kelker@isg-one.com

Abstract

Increasing speed and flexibility is of strategic importance to almost any company in times of digital transformation. While startups or "born digital" companies are agile by nature, traditional companies are struggling with the question of how to increase organizational agility. Little knowledge exists about how enterprises adopt and scale agile practices and structures. This exploratory study with twelve global cases examines how traditional companies adopt and scale agile structures. We found that (1) agile structures are currently adopted by enterprises at large scale, (2) agile structures are adopted not only by IT, but also by business units, and (3) while Spotify's organization serves as a widespread template for a fully agile unit, enterprises adapt and fine-tune this template according to their needs and scale. Furthermore, we identified three additional models for fully agile structures where a fully agile unit with cross-product support is the most frequently observed model.

1. Introduction

Digital transformation is ubiquitous and companies of almost all industries and size classes are under pressure to innovate on business models as new competitors create new products or services with the help of digital technologies [1, 2]. Digital transformation therefore imposes the need to react to rapidly changing market demands by sensing environmental change and responding readily [3, 4]. Enterprises adopt and scale agile practices to increase speed and flexibility, and, thus to increase agile capabilities [5].

Agile practices are inherently linked to software development and were initially considered to be only suitable for small and co-located software development teams [5]. The need to scale agile practices and their corresponding organizational structures beyond IT at the entire organization to tackle digital transformation

has emerged [6, 7]. Enterprises are now on the edge of adopting not only agile practices but also agile structures to increase organizational agility [5]. Consequently, organizations are successively moving away from transitional structures like bimodal IT where only parts of the organization are organized according to agile design principles towards structures where the entire organization follows fully agile structures [8].

Despite the growing agility literature [9-11], research on how traditional companies adopt and scale agile structures calls for a deeper understanding of (1) the application of agile practices and structures outside of software development [12], (2) the applicability of agile practices beyond small and co-located development teams [11], (3) the impact of adopting agile practices and structures at enterprises [13], and (4) how organizations can be structured to maximize benefits of adopting agile practices and structures [14].

This study aims at responding to this call for research with the following research questions:

- (1) *Which forms of agile structures can be observed at enterprises?*
- (2) *How do enterprises adapt generic agile structures to match their needs and scale?*
- (3) *Which migration paths can be observed at enterprises for the adoption of agile structures?*

While the adoption of agile practices or scaled agile frameworks like LeSS or SAFe is out of scope of this study, we focus on generating insights on how traditional enterprises adopt and scale agile structures by drawing on a multiple-case study with twelve cases from global companies across different industries.

2. Background

This section introduces the main theoretical concepts which are relevant for this study: Therefore, we refer to related research on agile software development, bimodal IT, IT and enterprise agility, and generic elements of agile units. Finally, we introduce Spotify's agile organization which serves as widespread template for a fully agile unit at startups or "born digital"

companies and increasingly also at traditional companies.

2.1. Agile software development

Agile practices and structures are closely related to IT due to their roots in software development. They originated as a response to challenges stemming from the traditional way IT is organized following "Plan-Build-Run" and the resulting separation between build and run [15]. Agile practices root in systems thinking and lean practices [16]. Systems thinking is about changing our perspective to solve problems in new and unexpected ways [17]. The Agile Manifesto is perceived as a first comprehensive collection of agile practices and aims at designing "better ways of developing software by doing it and helping others do it" [15]. It applies principles of systems thinking to software development: Individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan [15].

Agile software development is characterized by the following practices: Formulation of value stories, removing complexity, shortening release cycles to incorporate immediate customer feedback, and the estimation of built-effort with story points to reduce effort estimation complexity [18]. Agile practices aim at clean code, pair programming and immediate feedback, test-driven development, continuous integration, and automated testing [16]. Introducing agile practices in the IT function alone is not sufficient and requires "a more holistic approach [...] than one which is merely focused on continuous integration of software" [5] to increase agility in the entire organization. Consequently, the benefits of introducing agile practices and structures will be sub-optimal if not complemented by an agile approach in related functions outside IT [19].

As organizations scale, so do IT development and operations units. While they may initially be co-located with close communication links, increased team size and more strict separation of responsibilities can weaken such links [20]. Practitioners made several attempts to scale agile practices to the enterprise level: LeSS (Large Scale Scrum) is a lightweight agile framework developed by Craig Larman and Bas Vodde for scaling Scrum to more than one team [21] and SAFe (Scaled Agile Framework) is another approach developed by Dean Leffingwell for lean agile thinking and more visibly incorporating of scalable DevOps [6, 22]. A variety of agile practices has emerged with Extreme Programming, Kanban, Lean Startup, LeSS, SaFE, and Scrum as the most prominent [23].

2.2. Bimodal IT and enterprise agility

The term bimodal IT was initially coined by practitioners and is related to organizational ambidexterity [24]. Organizational ambidexterity is the ability of simultaneously pursuing exploitation and exploration [25]. Exploration is related to innovation capabilities and to "recombine potential resources in novel ways to create new capabilities and opportunities" [26] whereas exploitation is related to the efficient leverage and refinement of existing resources through known processes [26, 27]. Against this background, bimodal IT relates to ambidexterity through the ability of managing two separate but coherent working styles: One focusing on exploration, the other on exploitation [28]. Companies engage in bimodal IT to increase IT agility, IT exploratory capabilities, and the need for a structured business-IT alignment [29, 30].

In summary, we understand agility as a multidimensional concept [11, 31] where speed [32] and flexibility [3] are key elements. Consequently, enterprise agility can be defined as "ability of firms to sense environmental change and respond readily" [4].

2.3. Generic elements of fully agile units

This section briefly introduces basic agile concepts. The smallest unit of a fully agile structure – an agile team – is called "squad" and is outlined in Figure 1.

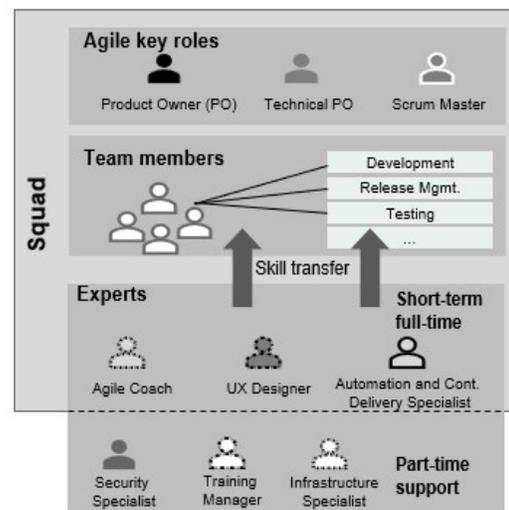


Figure 1. Basic agile team layout: The squad

A squad is designed like a 'mini startup' and has overall product responsibility [33, 34]. A squad has all required resources to design, develop, test, and deploy features and is a small cross-functional team. Squads

usually consist of eight to twelve permanent team members [18] and cover the following agile roles:

Product owner (PO): The PO represents the customer and ensures that the product delivers business value. He acts as customer and prioritizes work. The PO defines and accepts the product's features.

Technical product owner (TPO): The TPO supports the PO to ensure that the product delivers business value. He substitutes the PO. He has the overall technical responsibility for the product.

Scrum master: The Scrum master is responsible for ensuring that Scrum is understood and enacted and facilitates its methodology by coaching.

Agile coach: The agile coach is partly trainer and partly advisor helping agile teams to learn, apply, and to excel at agile practices. The agile coach usually serves several squads.

Team members: Design, build, test, integrate, maintain, and operate the product.

Experts: Contributor roles typically supported by external specialists on part-time or short-term basis in specific tasks where the squad lacks competencies.

Finally, a "tribe" is a group of squads with similar business interest and responsibility for a product area consisting of several related products. A tribe consists usually of eight to twelve squads and therefore contains up to 100 to 150 team members [18].

2.4. Model 1: Spotify-template/fully agile unit

This paragraph introduces a template for a fully agile structure that has been initially designed by startups and "born digital" companies like Spotify [33, 34]. This model as applied by Spotify has been described first by Kniberg (2012) and has been further elaborated by Gonçalves and Lopes (2014). While Kniberg focuses on agile structures applied by Spotify, Gonçalves and Lopes focus on agile practices and how LeSS has been implemented and adopted by Spotify. We refer to this template as 'model 1', a fully agile unit built on products.

Figure 2 shows model 1: One or more squads represent a product while a tribe consists of a product area covering several products or a product family [34]. A squad is led by the product owner who has overall product responsibility and represents the product towards the customer [34]. Each squad consists of all required resources to cover the entire product value chain [33, 34]. This typically includes the agile roles as outlined in Section 2.3 above.

Figure 2 further contains chapters and guilds: Chapters promote team collaboration and innovation and ensure methodological consistency across squads or tribes [33]. Chapters usually form around functional skills and the chapter lead often serves as functional line manager for chapter members [34]. Guilds are

communities of members with shared interest; they are less formal than chapters and represent a unit for informal exchange and knowledge sharing around a topic of interest [34].

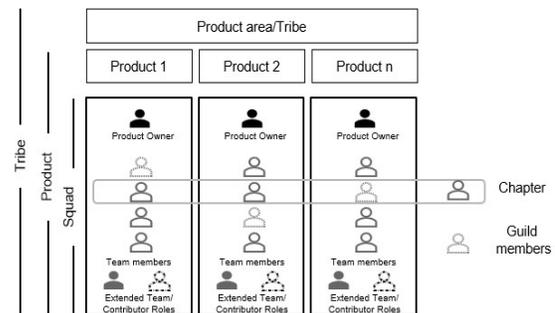


Figure 2. Model 1 – a generic agile unit as applied by Spotify

Despite its origin at startups and "born digital" companies, model 1 is in the meantime also popular at traditional companies due to its simplicity and ease of customization [18, 23]. This might be because model 1 is not limited to IT departments, but can also be applied to almost any other business unit with clear responsibility for a product or service [34].

Model 1 comes with the following advantages: First, squads have all required resources to make product-related decisions. This so-called product-aligned delivery speeds up decisions and implementation significantly because all decisions can be made within the squad [5]. Second, the issue of 'functional silos' is reduced since teams consist of all required resources to cover the product's value chain [35]. Third, team members have a mutual interest to 'not throw deliverables over the fence' since all squad members would suffer from a faulty product [18].

The disadvantages of model 1 are as follows: While squads would ideally have all required resources for product delivery, squads are usually confronted with limited resources and often do not have all required resources available since they depend on specialists for specific needs [16, 18]. Furthermore, full autonomy of the squads for all product-related decisions include complete freedom also regarding IT-architecture or the employed DevOps toolchain. This might lead to a lack of standardization and synergies across products [16].

3. Research methodology

This study aims at analyzing and comparing the adoption of agile structures at traditional enterprises. We have chosen an exploratory case study design with twelve global cases to maximize the chances of credible

novelty [36] and to allow for cross-case analysis to shed light on various organizational configurations applied by the case study companies [37, 38].

The case study companies were selected based on four criteria: First, to represent various industries and size classes to avoid potential bias. Second, we aimed also for business units applying agile structures as it is the case with CarCo-Drive, parts of RetailCo, and VehicleCo to identify possible differences in agile structures compared to IT departments. Third, we selected cases to which we had sufficient access to the case study companies to explore this novel phenomenon of interest in required depth. Finally, only companies have been considered that already applied scaled agile frameworks and structures like LeSS or SaFE as it is the case with the model 1 (i.e. the Spotify template as outlined above), currently one of the most widespread templates for a fully agile structure [33, 34]. Out of 19 companies that were initially identified from an outside-in-perspective as potentially relevant, twelve companies could be identified to fulfill this criterion.

Case study insights were derived in personal interviews – a method which is recommended in exploratory research to allow comprehensive discussions [36]. In each company, a minimum of one

senior manager (e.g. department/unit head) and an employee from the operative level has been identified to get a diverse view on how agile structures have been implemented. Additionally, executives and consultants facilitating agile transformation have been interviewed to further triangulate our findings.

In total 42 interviews have been conducted between November 2016 and April 2018 in either English or German. The interviews were conducted based on a semi-structured questionnaire and questions were mainly open-end to allow the interviewees the possibility to explore their experience and views in detail [37]. Follow-up questions have been formulated for further clarification purposes. Each interview had a duration of 60-120 minutes and was carried out primarily personally in face-to-face meetings. If further details were required, additional interviews have been conducted by telephone/Skype. The interview results were documented in detail in form of interview notes and, if permitted, in form of recorded interviews. The interviews were coded and reviewed for consistency and completeness by another researcher that has not participated at the interviews.

Table 1 lists details of the case study companies and conducted interviews.

Table 1. Overview and specifics of case study companies and conducted interviews

Case ID	Industry	Headquarter location	Size ¹	Company age [years]	Department	Interviews [#]	Interviewees' position ²
"CarCo-Drive"	Automotive	Germany	100+	100+	Business	8	(2); (3)
"CarCo-IT"	Automotive	Germany	100+	100+	IT	7	(1); (2); (3)
"Chemco"	Chemistry	Germany	100+	100+	IT	5	(1); (2); (3)
"EnergyCo"	Energy	Germany	40	100+	IT	2	(2); (3)
"FinCo"	Bank	UK	65+	100+	IT	2	(2); (3)
"InsureCo"	Insurance	Switzerland	4	100+	IT	3	(2); (3)
"RetailCo"	Retail	Germany	50	50+	IT; Business	3	(1); (2); (3)
"ServicesCo"	Services	USA	16+	90+	IT	2	(2); (3)
"SoftwareCo"	IT	USA	115+	40+	IT	1	(2)
"TelCo"	TelCo	Switzerland	17	20+	IT	4	(2); (3)
"ToolsCo"	Manufacturing	Liechtenstein	25	70+	IT	2	(2); (3)
"VehicleCo"	Automotive	Germany	100+	100+	Business	3	(2); (3)

1) in '000 [employees]

2) (1) Executive level, e.g. CIO, CDO, CTO; (2) Manager level, e.g. unit head, product owner, area product owner; (3) Agile team member, consultant, agile coach.

4. Results

Based on the findings of this case study we present in the following section three modifications of model 1 that we could observe during our research at the case study companies. We refer to these models as model 2, 3, and 4 (see Figures 3-5).

4.1. Model 2: Fully agile unit with cross-product support

Model 2 addresses the disadvantages of model 1 by adding shared cross-product functions to a tribe. Squads usually do not have all resources required for product delivery since they depend on specialists for tasks that only temporarily occur. In a scaled enterprise context this is especially true for alignment with non-product-related cross functional units like Finance & Controlling, HR, Purchasing, Legal, etc. that are usually organized non-agile. Furthermore, model 2 adds a cross-product architecture tribe to facilitate architectural consistency across feature teams.

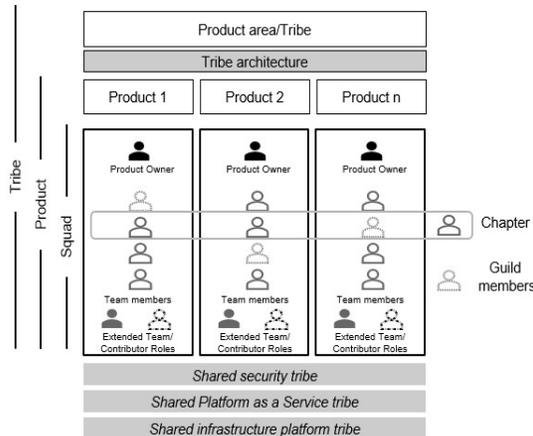


Figure 3. Fully agile unit (model 2 – with cross-functional tribes)

The differences of model 2 compared to model 1 are shown in Figure 3 in grey. FinCo as an industry leader in adopting agile structures was the first of the case study companies adopting model 2 already back in 2012. Model 2 has been adopted with some minor modifications also by InsureCo, ServicesCo, SoftwareCo, TelCo, ToolsCo and partly RetailCo to address issues of shared resources and to ensure standardization across tribes: A 'tribe architect' ensures that all squads follow the same architectural standards, use the same tools for agile software development and practices like Scrum or LeSS. Similarly, shared services tribes offer their services across squads to ensure efficient resources allocations.

Observed advantages of model 2 include an optimized resources allocation of shared services across products and a higher standardization. We observed especially at FinCo that a cross-product tribe architect facilitates sanity across the product landscape regarding the employed toolchain for agile software development as the cross-product tribe architect has the ultimate power of decision with respect to architectural questions. Furthermore, the area product owner is responsible for consistent product features across all products of the related product area and ensures an overall seamless customer experience.

Model 2 comes with the disadvantage that alignment with cross-functional/shared services tribes might slow down delivery due to the required alignment with shared services tribes. We observed related issues at FinCo, InsureCo, RetailCo, ServicesCo, and TelCo which were in constant fine-tuning regarding which services to be allocated in shared services tribes or squads.

4.2. Model 3: Fully agile unit with cross-product projects

Model 3 – as outlined in Figure 4 – further specifies model 2 and takes the existence of projects in a fully agile setting into account. Differences to model 2 are shown in shaded grey.

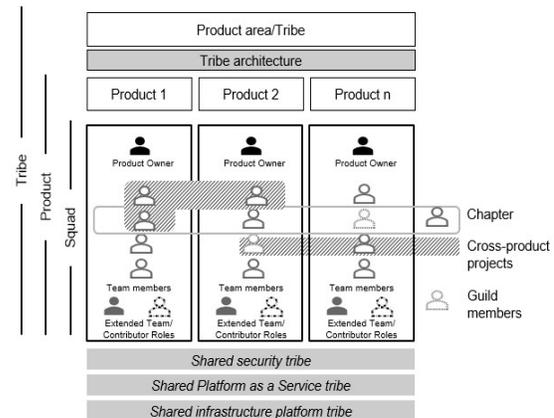


Figure 4. Fully agile unit (model 3 – additionally with cross-product projects)

The question of how to handle projects in a fully agile setting was subject to consideration at CarCo-IT and ChemCo. We understand a project as a temporary organization being unique, novel and transient [39]. In this sense, a project has a defined objective, a start- and end-date and is managed centrally by a dedicated project manager involving resources from various units. In a non-agile organization, projects play a key role for innovation and are comparatively easily handled: All involved units contribute with project resources

dedicated to the project with a specific proportion of their working time.

We observed conflicts between products and projects at model 3 that were similar to governance issues at matrix organizations where a unit has both, a disciplinary (e.g. industry or region), and functional reporting line. Matrix organizations usually resolve reporting line conflicts by defining a solid and a dotted reporting line clearly indicating which reporting line is leading. These conflicts are hard to be resolved within an agile organization if the project manager's priorities are in conflict with priorities of the product owner.

Both, CarCo-IT and ChemCo tried to handle resources prioritization conflicts between products and projects by assigning dedicated squad capacities to projects and treated projects like backlog items. This approach worked comparatively well as long as project resources demand was foreseeable and required little adaptation during the course of the project.

Frequent changes of project resources demand – as it is not uncommon for projects dealing with innovation and novelty – led to numerous capacity adjustments regarding squad team members involved in projects. These frequent resource alignments between the project manager and the product owner resulted in slowed down delivery of both, projects and squads. Consequently, CarCo-IT returned to model 2.

To avoid prioritization conflicts between products and projects, we have observed that FinCo, InsureCo, SoftwareCo, and ToolsCo avoided the initiation of projects at all and consequently classified topics predestined for projects as user stories (i.e. functional requirements) or backlog items. We observed another alternative at EnergyCo and ServicesCo where projects were delivered exclusively by external resources according to a fixed price and thus avoiding resource conflicts between projects and feature teams completely.

4.3. Model 4: Fully agile unit in a multiple-partner setting

Model 4 – as outlined in Figure 5 – was at the time of research only in place at CarCo-Drive. It further extends model 2 with multiple-partner delivery and an organizational separation between the development and operations function.

This model reflects specifics of CarCo's car engineering unit responsible for the development of an autonomous driving development platform. CarCo, a German premium original equipment manufacturer in the automotive industry (OEM), took on early initiative in developing own autonomous driving capabilities while simultaneously engaging in strategic partnerships with multiple partners including other OEMs and

original equipment suppliers (OES). These partnerships aimed at leveraging broader access to cutting-edge technologies like, for instance, machine learning and high-performance computing for the development of the autonomous driving platform.

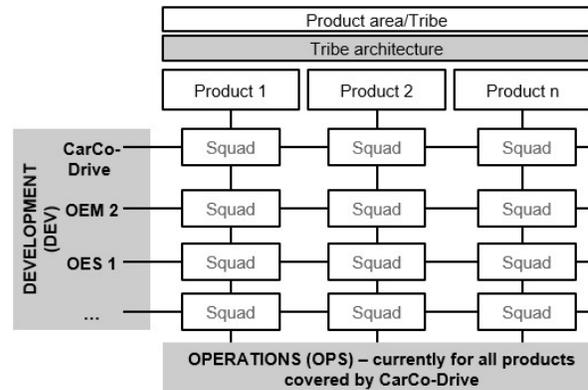


Figure 5. Fully agile unit in multiple-partner setting as applied by CarCo-Drive

Model 4 addresses the challenge of feature co-development for the platform: While the platform will be used by several OEMs and OESs, responsibility for development of features is clearly allocated to one OEM or OES and is organized according to the LeSS framework. At the time of our research, responsibility for operations for all features regardless of responsibility for feature development lay exclusively with CarCo-Drive reflecting an observation that agile practices are more relevant for development compared to IT operations [16, 18]. To avoid potential frictions between development and operations as intended in DevOps, IT operations resources from CarCo-Drive partly joined the development squads to ensure mutual knowledge exchange between the team members.

Model 4 extends model 2 as a fully agile unit with a multiple-partnership setting across companies. Model 4 allows for a further specialization of involved OEM/OES in feature development according to their core competencies while the clear allocation of operations to CarCo-Drive ensures high standardization and efficient platform operations across feature teams.

Model 4 comes with the shortcoming that the split of responsibility between development and operations in feature teams is prone for conflicts and inefficiencies: Despite of being part of the squads, CarCo-Drive resources lacked in part a mandatory understanding for features developed by partners. Consequently, CarCo-Drive struggled to realize synergies from operations across-products.

4.4. Migration paths towards fully agile organizations

Figure 6 provides an overview of models for fully agile structures we observed at the case study companies during our research and shows related migration paths.

Four out of twelve case study companies applied bimodal IT before selecting model 1 as initial starting point for a fully agile unit.

Model 1 remained as resulting fully agile structure in three cases: EnergyCo (IT department; initially bimodal), VehicleCo (business unit, initially not bimodal), and partly RetailCo (business unit, initially not bimodal). While VehicleCo and partly RetailCo represent rather self-sufficient business units providing business analytics services, EnergyCo represents the IT department of a recently founded spin-off for green energy of a German energy incumbent. These organizations are of reasonable size with limited alignment need across products or large shared service tribes serving several product areas.

With eight out of twelve cases, the majority of the case study companies, CarCo-IT, FinCo, InsureCo, ServicesCo, SoftwareCo, TelCo, ToolsCo, and partly RetailCo adopted model 2 as resulting state of a fully agile structure during our research.

CarCo-Drive, CarCo's autonomous driving business unit is the only case where an organization migrated from model 2 to model 4 taking the specifics of development in a multi-partnering setting into account.

Model 3, a fully agile unit with cross-product tribes for architecture and shared services in a joint setting with projects has been adopted by CarCo-IT and ChemCo after adopting model 1. While ChemCo remained at the time of research the only company organized according to model 3, CarCo-IT eliminated projects completely and adopted model 2. Noteworthy, model 3 has only been adopted by cases that were initially bimodal: Startups or "born digital" companies usually immediately turn into fully agile structures skipping bimodal settings completely [40]. Contrary, traditional companies like CarCo, ChemCo, or RetailCo – all in business for at least 50 years – came historically from a non-agile environment where projects played an important role for fostering innovation, dealing with novelty, or managing organizational change.

Not surprisingly, companies more inclined to agile practices and structures like SoftwareCo, CarCo-Drive, or the business analytics units at RetailCo or VehicleCo adopted fully agile structures right from the beginning. These companies eliminated projects completely.

While the majority of our cases represent IT organizations, three out of twelve cases represent business units: CarCo-Drive, VehicleCo, and partly RetailCo. Again, all of them immediately adopted a fully agile structure and skipped bimodal stages confirming findings of other researchers like Horlach et al. that bimodal IT "still mainly implies the transformation of the IT organization and does not focus on transforming the whole organization" [30].

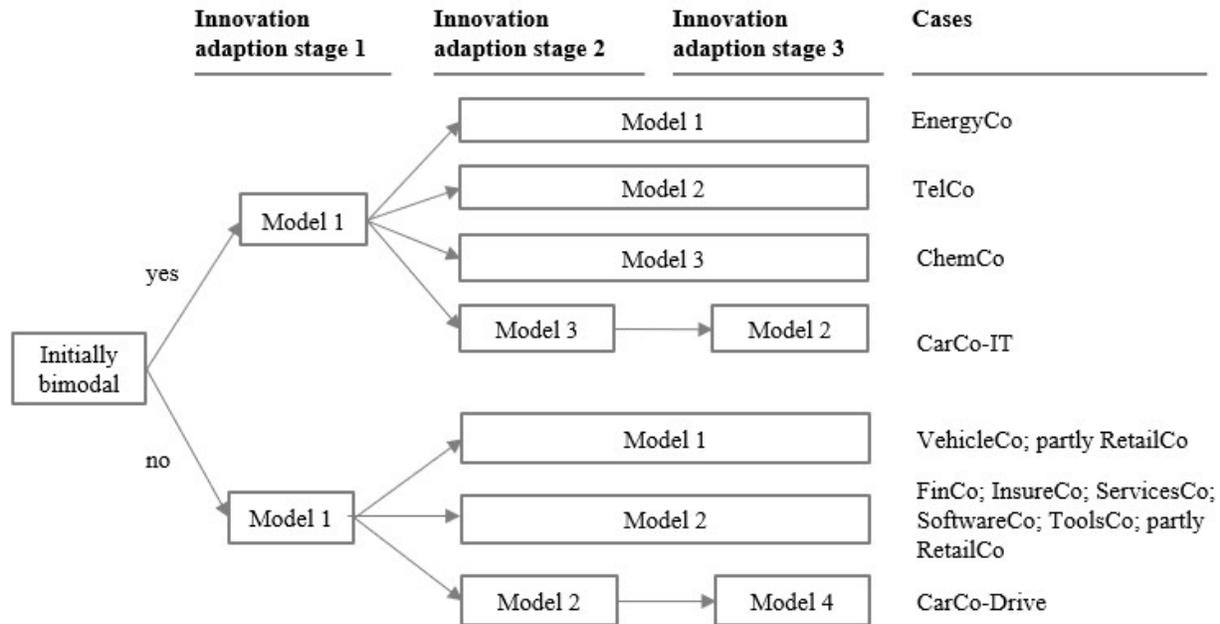


Figure 6. Observed migration paths of agile structures at case study companies

5. Discussion

While bimodal IT dominates the extant literature related to agile structures in IS research [41, 42], our case study shows that just four out of twelve case study companies initially applied bimodal IT – CarCo-IT, ChemCo, EnergyCo, and TelCo. All four companies have replaced bimodal IT by fully agile structures. Other publications confirm this observation [8] and a recent survey reveals that 63% of the respondents do not trust bimodal IT anymore while 90% perceive cross-functional teams and agile structures as superior [8].

While bimodal IT appeared just recently [28], agile structures are not new [9]: Industry leaders like FinCo, RetailCo, SoftwareCo or ToolsCo made first steps towards agility almost ten years ago. While applying agile practices and structures is the normal modus operandi for startups or "born digital" companies [40], FinCo and ToolsCo are especially noteworthy: FinCo is an industry pioneer in the financial services industry regarding the adoption of agile structures and used digital technologies early for product innovation. Likewise, ToolsCo innovated its business model by applying digital technologies from selling drilling tools to selling drilling as a service resulting in recurring revenue and new service offerings like predictive tools maintenance or automatic supplies replenishment.

Our case study confirms the findings of other researchers that, as companies innovate on business models and develop new products or services with the help of digital technologies [2], new business units are predestined to be established as fully agile structures [41, 42].

The three cases representing business units, CarCo-Drive, VehicleCo and partly RetailCo, are excellent examples for business units organized according to agile practices and structures as new, digital business ventures outside IT departments: The global car manufacturer VehicleCo has established an analytics unit as a fully agile structure and RetailCo established a new business venture outside IT. In a similar fashion, CarCo, has established a fully agile unit as part of their car development department to establish a development system for fully autonomous driving of level 4 (fully automated) and level 5 (driverless). As of October 1st, 2017, CarCo has decided to reorganize the autonomous driving unit consisting of more than 800 employees according to the agile framework LeSS and in a fully agile setting creating CarCo's first fully agile unit outside IT.

These are exemplary cases where fully agile structures have been adopted by business units outside IT. All cases have in common that their business model heavily relies on digital technologies [2]. At all case study companies, digital technologies are provided out of fully agile units rather than non-agile or bimodal IT departments. Agile structures therefore have clearly left behind the experimental stage only relevant for startups or strategically non-relevant small units [5]. Consequently, adopting agile practices and structures not only in IT but also in business units is required to increase flexibility and speed of the entire organization [6, 19]. If an organization wishes to be truly agile, its software teams cannot be islands of agile practices and structures – rather the entire organization needs to embrace agility in its processes [43].

Figure 7 summarizes the results of our case study.

Model	Description	Cases	Advantages	Disadvantages	Delivery description
1	Generic fully agile unit	EnergyCo; VehicleCo, partly RetailCo	<ul style="list-style-type: none"> – Squads have all required resources – Avoids 'functional silos' – Mutual interest to provide deliverables 'end to end' 	<ul style="list-style-type: none"> – Resources limitations, especially regarding specialists – Lack of synergies and standardization 	<ul style="list-style-type: none"> – Squad responsible for one product – PO prioritizes the features – Product development and operations done by the same feature team
2	Fully agile unit with cross-functional tribe	CarCo-IT; FinCo; InsureCo; ServicesCo; SoftwareCo; TelCo; ToolsCo; partly RetailCo	<ul style="list-style-type: none"> – Optimized resources allocation due to shared resources – Cross-functional tribe ensuring product/ architecture consistency across products 	<ul style="list-style-type: none"> – Cross-product alignment might slow down product delivery 	<ul style="list-style-type: none"> – Additionally to model 1: <ul style="list-style-type: none"> – Cross-functional tribe ensures efficient allocation of specialized resources and decides about architecture and tools
3	Fully agile unit with cross-product projects	ChemCo	<ul style="list-style-type: none"> – Reflection of projects in a fully agile setting 	<ul style="list-style-type: none"> – Prioritization conflicts between the product and project 	<ul style="list-style-type: none"> – Squad takes care for delivery of product features – Project covers additional topics
4	Fully agile unit in a multiple-partner setting	CarCo-Drive	<ul style="list-style-type: none"> – Reflection of development in a multiple-partner setting – Split of responsibility between Dev and Ops – Operations synergies realized 	<ul style="list-style-type: none"> – Potential disconnect between Dev and Ops – Increased complexity – Governance challenges with multiple-partners 	<ul style="list-style-type: none"> – Multiple delivery units taking care for product feature development – Ops unit separated from the dev unit allowing the ops unit to specialize on operations exclusively

Figure 7. Summary of case study findings

6. Conclusion

The question of how to increase speed and flexibility to be as adaptable and resilient while maintaining efficiency and reliability is of strategic importance to almost any company in times of digital transformation [2, 3]. While previous research primarily focused on agile practices and structures as software development method or on bimodal IT [29, 30, 41, 42], our research is motivated by the lack of empirical evidence on how traditional enterprises adopt and scale agile structures.

An exploratory study with twelve global cases has been conducted to gain a deeper understanding of the current state of adopted agile structures at enterprises. We found that companies start their agile transformation by adopting a template for a generic fully agile unit regardless of whether the initial setting was bimodal or not. With increased agile maturity stages, companies enhance this template to their needs by incorporating shared service tribes. We found that innovative business units were more open towards directly adopting fully agile structures and skipping an initial bimodal setting.

Our case study reveals that the adoption of agile structures currently takes place at enterprises at large scale regardless of industry or size.

We build on extant research related to bimodal IT and enterprise agility and further extend it to specifically explore the adoption of agile structures at enterprises. We especially extend the existing work on the adoption of agile structures in IT and the development of bimodal IT by confirming findings of other researchers that digital transformation leaders aim at fully agile structures [24, 29, 30, 41].

This study has several practical contributions: We have extended the so called 'Spotify-Template' for a fully agile unit (model 1) by three others (model 2-4) taking specific needs of enterprises regarding scale and scope into account. Companies that haven't adopted fully agile structures yet could use these models as references for designing agile structures. Companies

that have already implemented agile structures could use the identified models for reflection with their own experience.

Our study does not come without limitations: We have identified twelve cases to be as representative as possible for traditional enterprises that have already started adopting agile structures. These cases might not be fully representative for companies of all industries or sizes. Furthermore, except for FinCo, RetailCo, SoftwareCo, and ToolsCo, the case study companies were at a comparably early stage of adopting agile structures resulting in still ongoing changes regarding organizational setup. Consequently, the presented agile structures are snapshots of the current state of agile transformation during time of our research with a high likelihood that adopted agile structures will be further modified over time. Finally, this multiple case-study relies on the information provided in the interviews by a comparatively small number of total respondents (42 interviews in total).

Future work should specifically extend to: (1) Examining the adoption of agile structures in a longitudinal perspective, (2) understanding how agile structures are adopted not only in IT but also in business units that are not engaged in IT development or delivery, (3) understanding the challenges created during the transformation process from introducing agile structures where agile and non-agile units coexist, (4) understanding how optimization could be realized by adopting agile structures, and (5) exploring how the scaling of agile structures could potentially unlock enterprise agility as an organizational capability.

Despite the imposed challenges, our research indicates that the adoption of agile structures is more than a short-term, transitory trend and will play a significant role as companies need to increase speed and flexibility to innovate with new digital products and services. It remains striking to learn how agile structures will be adopted by enterprises in IT and in business units as they move from "doing agile" to "being agile".

7. References

- [1] Kohli, R., and Melville, N.P.: 'Digital innovation: A review and synthesis', *Information Systems Journal*, 2018, pp. 1-24.
- [2] Weill, P., and Woerner, S.L.: 'Thriving in an increasingly digital ecosystem', *MIT Sloan Management Review*, 2015, 56, (4), pp. 27-34
- [3] Highsmith, J.A.: 'Agile project management: creating innovative products' (Pearson Education, 2009)
- [4] Overby, E., Bharadwaj, A., and Sambamurthy, V.: 'Enterprise agility and the enabling role of information

- technology', *European Journal of Information Systems*, 2006, 15, (2), pp. 120-131
- [5] Fitzgerald, B., and Stol, K.-J.: 'Continuous software engineering: A roadmap and agenda', *Journal of Systems & Software*, 2017, 123, pp. 176-189
- [6] Leffingwell, D.: 'Scaling software agility: best practices for large enterprises' (Pearson Education, 2007. 2007)
- [7] Reifer, D.J., Maurer, F., and Erdogmus, H.: 'Scaling agile methods', *IEEE Software*, 2003, 20, (4), pp. 12-14
- [8] Roemer, M., Weiss, C., Venus, M., Linhart, A., Eistert, T., Schmid, J., Roeglinger, M., and Utz, L.: 'Designing IT Setups in the Digital Age' (Fraunhofer FIT, 2017)
- [9] Conboy, K.: 'Agility from First Principles: Reconstructing the Concept of Agility in Information

- Systems Development', *Information Systems Research*, 2009, 20, (3), pp. 329-354
- [10] Wang, X., Conboy, K., and Pikkarainen, M.: 'Assimilation of agile practices in use', *Information Systems Journal*, 2012, 22, (6), pp. 435-455
- [11] Abrahamsson, P., Conboy, K., and Wang, X.: '"Lots done, more to do": the current state of agile systems development research', *European Journal of Information Systems*, 2009, 18, (4), pp. 281-284
- [12] Kiely, G., Kiely, J., and Nolan, C.: 'Scaling Agile Methods to Process Improvement Projects: A Global Virtual Team Case Study'. *Proceedings of the 23rd Americas Conference on Information Systems*, Boston, MA, USA2017 pp. 1-9
- [13] Gerster, D., Dremel, C., and Kelker, P.: '"Agile meets non-agile": Implications of adopting agile practices at enterprises'. *Proceedings of the 24th Americas Conference on Information Systems*, New Orleans 2018
- [14] Maruping, L.M., Venkatesh, V., and Agarwal, R.: 'A control theory perspective on agile methodology use and changing user requirements', *Information Systems Research*, 2009, 20, (3), pp. 377-399
- [15] Fowler, M., and Highsmith, J.A.: 'The agile manifesto', *Software Development*, 2001, 9, (8), pp. 28-35
- [16] Kulak, D., and Li, H.: 'The Journey to Enterprise Agility: Systems Thinking and Organizational Legacy' (Springer, 2017)
- [17] Deming, W.E.: 'The new economics: for industry, government, education' (MIT press, 2000. 2000)
- [18] Kim, G., Debois, P., Willis, J., and Humble, J.: 'The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations' (IT Revolution, 2016)
- [19] Overby, E., Bharadwaj, A., and Sambamurthy, V.: 'A Framework for Enterprise Agility and the Enabling Role of Digital Options', in Baskerville, R.L., Mathiassen, L., Pries-Heje, J., and DeGross, J.I. (Eds.): 'Business Agility and Information Technology Diffusion' (Springer US, 2005), pp. 295-312
- [20] Swartout, P.: 'Continuous Delivery and DevOps – A Quickstart Guide' (Packt Publishing Ltd, 2014. 2014)
- [21] Larman, C., and Vodde, B. 2017. "Less.Works." Retrieved 19.4.2018, 2018, from <https://less.works/less/framework/index.html>
- [22] ScaledAgile. 2017. "Essential Safe 4.5." Retrieved 29.10.2017, from <http://www.scaledagileframework.com/>
- [23] Versionone. 2018. "12th annual state of agile report." Retrieved 25.8.2018, 2018, from <https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report>
- [24] Joehnk, J., Röglinger, M., Thimmel, M., and Urbach, N.: 'How to implement agile IT setups: A Taxonomy of design options'. *Proceedings of the 24th European Conference on Information Systems (ECIS)*, Guimaraes, Portugal 2017, pp. 1521-1535
- [25] O'Reilly, C.A., and Tushman, M.L.: 'Ambidexterity as a dynamic capability: Resolving the innovator's dilemma', *Research in Organizational Behavior*, 2008, 28, pp. 185-206
- [26] Lee, O.-K., Sambamurthy, V., Lim, K.H., and Kwok Kee, W.: 'How Does IT Ambidexterity Impact Organizational Agility?', *Information Systems Research*, 2015, 26, (2), pp. 398
- [27] March, J.G.: 'Exploration and exploitation in organizational learning', *Organization Science.*, 1991, 2, (1), pp. 71-87
- [28] Bygstad, B. 2015. "The Coming of Lightweight IT". *Proceedings of the of 23rd European Conference on Information Systems*, Münster 2015.
- [29] Haffke, I., Kalgovas, B., and Benlian, A.: 'The Transformative Role of Bimodal IT in an Era of Digital Business', (2017), pp. 5460-5469
- [30] Horlach, B., Drews, P., and Schirmer, I.: 'Bimodal IT: Business-IT Alignment in the Age of Digital Transformation'. *Proceedings of the MKWI 2016* pp. 1417-1428
- [31] Holmström, H., Fitzgerald, B., Ågerfalk, P.J., and Conchúir, E.Ó.: 'Agile practices reduce distance in global software development', *Information Systems Management*, 2006, 23, (3), pp. 7-18
- [32] Lyytinen, K., and Rose, G.M.: 'Information system development agility as organizational learning', *European Journal of Information Systems*, 2006, 15, (2), pp. 183-199
- [33] Gonçalves, E., and Lopes, E.: 'Implementing Scrum as an IT Project Management Agile Methodology in a Large Scale Institution', pp. 461-470
- [34] Kniberg, H. I., Anders. 2012. "Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds." from <https://creativeheldstab.com/wp-content/uploads/2014/09/scaling-agile-spotify-11.pdf>
- [35] Ambler, S.W.: 'Scaling agile: an executive guide', (IBM, 2010, edn.), pp. 1-21
- [36] Langlely, A., and Abdallah, C.: 'Templates and turns in qualitative studies of strategy and management': 'Building methodological bridges' (Emerald Group Publishing Limited, 2011), pp. 201-235
- [37] Yin, R.K.: 'Case Study Research - Design and Methods' (Sage, 2009. 2009)
- [38] Eisenhardt, K.M.: 'Building Theories from Case-Study Research', *Academy of Management Review*, 1989, 14, (4), pp. 532-550
- [39] Turner, J.R., and Müller, R.: 'On the nature of the project as a temporary organization', *International Journal of Project Management*, 2003, 21, (1), pp. 1-8
- [40] Tumbas, S., Berente, N., and vom Brocke, J.: 'Born Digital: Growth Trajectories of Entrepreneurial Organizations Spanning Institutional Fields'. *Proceedings of the 38th International Conference on Information Systems (ICIS)*, Seoul, Korea 2017
- [41] Horlach, B., Drews, P., Schirmer, I., and Böhmman, T.: 'Increasing the Agility of IT Delivery: Five Types of Bimodal IT Organization'. *Proceedings of the 50th Hawaii International Conference on System Sciences*, Hawaii, USA2017 pp. 5420-5429
- [42] Haffke, I., Kalgovas, B., and Benlian, A.: 'Options for Transforming the IT Function Using Bimodal IT', *MIS Quarterly Executive*, 2017, 16, (2), pp. 101-120
- [43] Wendler, R.: 'Dimensions of Organizational Agility in the Software and IT Service Industry: Insights from an Empirical Investigation', *CAIS*, 2016, 39, pp. 439-482