

# Master Thesis Organizational Design & Development



An organizational design perspective on self-organizing potential and customer value in agile teams

*A qualitative case study at Unit4*

Author: Federica Della Rupe - 4618874

Supervisor: Dr. Matthijs Moorkamp

2<sup>nd</sup> examiner: Dr.ir. L.J. Lekkerkerk

## *Abstract*

We are living in a complex world, where to disrupt or being be disrupted seems to be a rule in the software industry. The only certainty for business is the increasing speed of change, with an extreme pressure to adapt to those changing circumstances as quickly as possible and to create valuable products and services for customers. This thesis reports on a qualitative case study at Unit4, an international software company that decided a few years ago to implement Agile in the R&D Benelux department. Three theories are combined in the theoretical framework of this master thesis to analyze the relation between self-organizing potential of agile teams and customer value, such as Agile, Lean and Sociotechnical design theory. The aim of this study is addressing the relation between self-organizing potential of agile teams and the creation of customer value in the R&D Benelux department, in an organizational design perspective. Considering that organizations that are facing increasing uncertainty and complexity need to invest in organizational redesign, an organizational design perspective is chosen. The assessment of agile team features, degree of autonomy and degree of Scrum implementations, shows a positive impact on the creation of customer value.

*Key words:* Self-organizing potential, Software, Autonomy, Agile, Scrum, Customer feedback, Customer involvement, Customer value, Sociotechnical design theory.

Abstract .....	2
1 Introduction .....	5
1.1 Research object and question .....	5
1.2 R&D Department, Benelux domestic.....	7
1.3 Practical relevance.....	9
1.4 Academic relevance .....	9
2 Theoretical background .....	11
2.1 Customer Value .....	11
2.1.1 Customer value - Lean.....	12
2.1.2 Customers give feedback on quality and delivery performance .....	13
2.1.3 Customers involvement .....	13
2.2 Self-organizing potential within Agile.....	14
2.2.1 Agile team features in software development .....	15
2.2.2 Degree of autonomy .....	16
2.2.3 Scrum framework – degree of Scrum implementation .....	17
2.3 Structural characteristics – Sociotechnical Design theory.....	18
2.3.1 Functional concentration .....	19
2.3.2 The level of differentiation of operational transformation.....	21
2.4 Relations between concepts.....	22
3 Methodology.....	23
3.1 Case study and systematic combining .....	23
3.2 Qualitative data collection – interviews and sample .....	24
3.3 Operationalizing and Analysis.....	25
3.4 Ethical considerations.....	26
4 Results.....	28

4.1	Axial phase .....	29
4.1.1	Working with a Continuous Delivery model .....	29
4.1.2	Dealing with the Development, Testing, Acceptance and Production (DTAP) model 35	
4.1.3	R&D working with other departments .....	37
4.2	Analysis .....	40
4.2.1	SOP and CV .....	41
4.2.2	SOP, CV and structural characteristics .....	43
5	Conclusion .....	46
6	Discussion .....	48
6.1	Limitations .....	48
6.2	Managerial Advice .....	49
6.3	Contribution to knowledge and future research .....	50
7	References .....	52
8	Appendix .....	56
8.1	Appendix 1 – Two exploratory interviews on Agile and R&D department at Unit4 .....	56
8.2	Appendix 2 – Operationalizing and Proposed interview guide .....	57
8.2.1	Operationalizing (examples of questions). .....	57
8.2.2	Proposed list of interviews questions for Agile teams .....	59
8.3	Appendix 3 – Informed Consent form for interviews .....	61
8.4	Appendix 4 – Coding (statements cards and analysis map) .....	62
8.5	Appendix – Scrum Roles and Unit4 R&D Benelux department .....	63

# 1 Introduction

*The case study analyzed in this thesis, is the result of the knowledge acquired during the Master in Organizational Design and Development at Radboud University, and the working experience at Unit4, a software company based in Utrecht, in which I have been working for one year and a half, as Organizational Development coordinator. I saw a precious opportunity to apply the theory and the knowledge acquired during this Master's degree, to a dynamic and changing organization. The self-organizing potential in the R&D department and its relationship with customer value, is an interesting topic to analyze in the final Master thesis project from an organizational design perspective.*

## 1.1 Research object and question

Unit4 is a leading provider of enterprise applications empowering people in service organizations. With annual revenues of 500M Euro and more than 3,100 employees world-wide, Unit4 delivers their own enterprise resource planning (ERP), industry-focused and best-in-class applications. Thousands of organizations from sectors including public services, real estate, education, wholesale, financial services and professional services benefit from Unit4 solutions. Unit4 used to be a quite fragmented organization, which has been experiencing a centralization process since March 2014 (Unit4 Blog, 2014); due to the private equity investment from Advent International, which is one of the largest and most experienced global private equity firms.

Unit4 is reshaping itself and the result of this transition is the implementation of a global infrastructure, designed to enable efficient growth and enhance productivity, and to improve the value produced for Unit4 customers. As part of this transition, Unit4 chooses to implement agile methods within the R&D department, which is impacting the way of working and the organizational culture and infrastructure.

Agility seems to be a “trend” word, which describes the readiness to rapidly or inherently create change, proactively or reactively embrace change, and learn from change, while contributing to perceived customer value, through its collective components and relationships with its environment (Conboy 2009). An agile methodology has been implemented since 2001 in many software organizations; for Unit4, the main purposes for this implementation were the improvement of customer value and the self-organizing potential at Unit4 in the R&D department. For this reason, in February 2017 agile methodology has been brought into 20 to 25 teams in the R&D department. Self-organizing potential is a strong component of agile methodologies, which

characterizes the way those teams are working and operating; Unit4 has been introducing several agile methods, such as Scrum and Kanban, which belong to the umbrella of agile methodology.

While with the implementation of agile methodology Unit4 made a big step towards the implementation of self-organizing potential and its impact on positively increasing customer value, the results of these efforts can be further improved. A constructive criticism towards agile methodology has been stated by Moe, Dingsøyr and Dybå (2008), who note that agile methodology emphasizes self-organizing teams but does not provide clear guidelines on how they should be implemented and how the organizational structure should be adjusted accordingly. Therefore, on one hand this thesis will investigate the relation between self-organizing potential of agile teams and the creation of customer value; while on the other hand, an organizational design perspective will be considered.

Literature indicates that for achieving self-organization in organizations, a more integral redesign perspective is relevant, such as the Sociotechnical design approach. For example, in *Lowlands Sociotechnical Design Theory and Lean*, the Sociotechnical design theory has been applied to Lean, to transform the success story of Lean in key work of organizational structure and organizational design (Christis & Soepenbergh 2015). De Sitter et al (1997) state that a more integral design approach enables organizations to develop and enables the members of the organization to develop and use their own design skills and expertise. The Sociotechnical design theory has been applied by dozens of Dutch firms and the implications of a design approach is reflected with a positive outcome for organization's development. *"Organizations that are confronted with increasing uncertainty and complexity have to invest in organizational redesign in order to survive"* (De Sitter et al., 1997, p.2). Those are just examples to mention the power and the benefits of a more integral redesign approach within an organization.

To facilitate Unit4's transformation process, this master thesis will investigate the relation between self-organizing potential and the creation of customer value from the perspective of organizational design. The structural characteristics of the Sociotechnical design theory will be considered as an influence; this will be further and deeper investigated in the next chapter of this thesis. The research question of this case study within Unit4 is:

*What is the relation between self-organizing potential and the creation of customer value and how is this influenced by structural characteristics, in R&D agile teams at Unit4?*

The research question can be further specified with the two following sub-questions, considering the relation between independent and depended variable and which structural properties will influence the outcome variables:

- 1) *What is the relation between self-organizing potential and the creation of customer value in R&D agile teams?*
- 2) *What is the degree by which the structural characteristics can accelerate this relation?*

## 1.2 R&D Department, Benelux domestic

The R&D domestic Benelux department is the “system in focus” of this study, which includes around two hundred R&D employees, who are responsible for many different software classified in five different lines of business. Understanding the layout of the R&D Benelux department was a crucial step before diving into the theoretical framework and into the data collection process. On top of the five lines of business, a General R&D team is taking care of the whole domestic lines of business; this team is the sponsor (project leader) of the Agile transformation project at Unit4, that was conducted in the past years. The Agile way of working has been introduced in the R&D Benelux department for facilitating a big reorganization that brought down the R&D Benelux organization from eleven lines of business to five domestic lines of business. As can be seen in figure 1, presenting the Organization chart of the R&D Benelux, the lines of business are the following, responsible for vertical markets:

1. *Health Care*
2. *Office of HR*
3. *Financial services*
4. *SME*
5. *PS and product service*

One of the reasons for Scrum teams was the introduction of one common way of working and an easier method for management to re-allocate employees from one line of business to another. Another important reason was getting closer to the customer and getting the teams closer to the customers, and this is the main reason for the selection of the R&D Benelux population to investigate the creation of customer value, making explicit the consideration that R&D employees are the main creators of the products that the organization is selling to the outside world. For this reason, they are as well responsible for the creation of high quality software, which is valuable for the customers of Unit4.

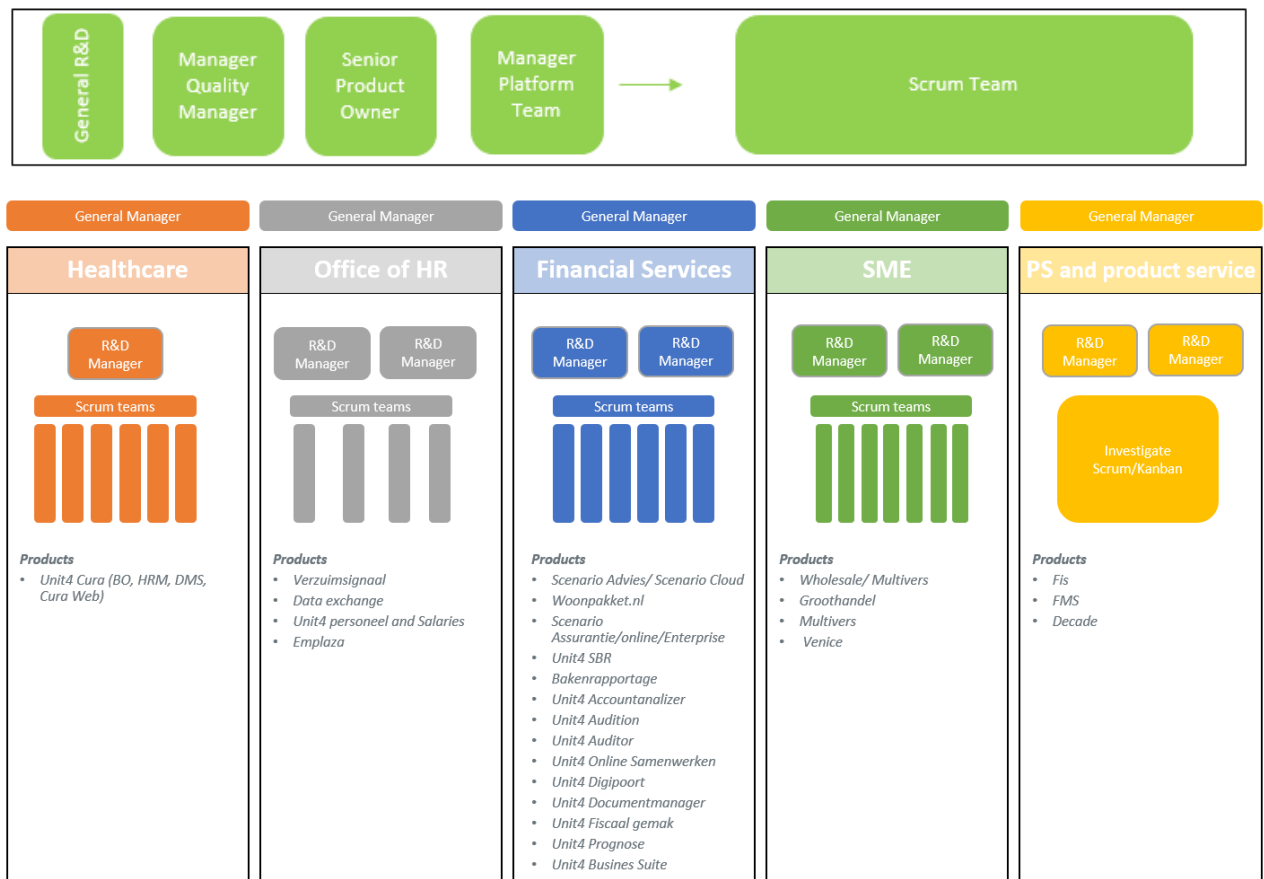


Figure 1: R&D Benelux Organisatie, Unit4 2018 (in Appendix 8.5 a bigger section is available)

At Unit4 the R&D Manager is called Product Owner, following the agile terminology, and he/she is responsible for the maximization of the return on investment (ROI) of the product and to foster the value of the work of the development team in general. At Unit4 the Product Owner is collaborating with a Product Manager and together they are the persons which manage the Product Backlog, which is a prioritized features list that includes the priorities and the activities for the team. The Development Team is a flat and self-organized Scrum team which is composed of developers and testers, who are responsible of creating the output of each Sprint, which is called increment in agile terminology. Even if individual team members may have specialized expertise, the whole team is accountable for turning the Product Backlog into a valuable product for the customers. At Unit4 the size of the teams can vary between three and nine members. The Scrum Master is responsible to facilitate the interactions between the Scrum Team and the external teams. His or her role is to ensure that Scrum theory, practices and rules are understood and applied in the correct way.

Scrum is not the only agile methodology that has been implemented at Unit4. Some self-organizing teams at Unit4 are working with another agile method, which is called *Kanban* in the



PS and products service line of business. This framework is lighter than Scrum, but still a strong incentive towards transparency. The job of the teams that are working with Kanban, is not related to new development, but is more maintenance of existing application. When addressing self-organizing potential, Scrum is much better defined and explicit around self-organizing rules, which are a must to implement the methodology. This is the reason why R&D Scrum teams are the “system in focus” in this Master thesis. The sample chosen for interviews will be addressed in the methodology section, later.

### 1.3 Practical relevance

Unit4 has been undertaking an agile transformation since February 2017. The R&D department was the department in which the agile transformation has been implemented and carried out. There have been different “waves” of agile transformation at Unit4. Every wave exists of four teams that will be trained and coached by Unit4 Agile Experts. Every wave has the same rhythm of training and coaching, and includes preparation work, from an organizational culture point of view and from assessment of the maturity of the teams involved. This master thesis is aiming at giving practical recommendations to the management team at Unit4, looking at the current status of the R&D Benelux department.

In a new business era, modern ERP software must ensure consistency and availability. The only certain for businesses is the increasing speed of change, with an extreme pressure on business to adapt to those changing circumstances as quickly as possible. To disrupt or to be disruptive, this is the current reality in the software industry. *“Technology became a catalyst for change and agility became the holy grail”* (Unit4 Blog, 2017). The Scrum alliance reports that 87 percent of teams improved their quality of work and their working life with this method (2015). The study of self-organizing potential (in an agile key and methods), structural characteristics at Unit4, and their impact on customer value, will help the organization in the upcoming waves of transformation. Furthermore, it will provide more insights into when Unit4 will be ready to work in an agile way within all the departments of the organization; for this purpose, structural considerations will be taken into account from a Sociotechnical design perspective.

### 1.4 Academic relevance

Agility in a software development context is a topic that has been largely discussed in literature; there are a lot of different examples and articles which address this topic in a context of software organizations. This study is adding value to the literature because it analyzes the topic

form an organizational design perspective, which has not been in focus that way before. Agility transformations are implemented and sold to organizations as a very simple and immediate design, applicable to every business reality; those considerations are most of the time poor in theory. This thesis project will take into account theory behind the agile design and structural characteristics and methods from Sociotechnical design theory. Moe et al. (2008) note that Scrum emphasizes self-organizing teams but does not provide clear guidelines on how they should be implemented and how the organizational structure should be adjusted accordingly.

The aim of the study is to extend the existing theory on agility, contributing to the extensive literature on this topic, by looking at a more organizational design perspective, and considering the structure of the organization.

## 2 Theoretical background

In this second chapter the main variables and concepts will be identified, which come from the main theories used for building this theoretical framework.

### 2.1 Customer Value

In a changing, dynamic and global market, being able for companies to create customer value, has been defined as the key for competitive advantage by many authors in recent years. Competitive advantage in markets via superior customer value delivery is a key factor that could make the difference between successful and mediocre organizations. Customer value-based competition has been defined as the next major shift in managerial practices and will require a new way of managing teams, internal processes and a new set of skills to marry internal quality with external customer value (Woodruff, 1997). But first a step back is necessary to identify several definitions of customer value which are available in literature.

The following quotes all represent relevant aspects of the customer value, and they have been relevant for the researcher to investigate the concept of customer value in literature. The most relevant quote is the one from Womack and Jones in 1997, because it is from a Lean perspective that the researcher uses to look at the creation of customer value, as it will be explained in the following paragraph.

*“Value is the consumer's overall assessment of the utility of a product based on perceptions of what is received and what is given” (Zeithaml 1988, p. 14)*

*“By customer value, we mean the emotional bond established between a customer and a producer after the customer has used a salient product or service produced by that supplier and found the product to provide an added value” (Butz and Goodstein 1996, p. 63)*

*“First, there is the need to specify value. Value, it is argued, should be defined by the customer, in terms of specific products with specific capabilities at specific prices. Second, the value stream should be identified. The value stream incorporates all the actions required to bring the product to the customer: including detailed design, engineering, production, order-taking, production scheduling and delivery. This stage should identify activities that add value, that do not add value but are unavoidable in current circumstances and those that do not add value and are avoidable. Those activities in the third category should be eliminated. The third stage is to create flow” (Womack and Jones 1997, p.1148)*

### 2.1.1 Customer value - Lean

A theory that has largely considered customer value is Lean, because in Lean's literature the two main goals have been identified as reducing waste and increasing customer value. Looking at history is necessary to understand the origins of Lean, which can be found at Toyota Motor Corporation, with innovations for the automobiles market including Just-in time (JIT) production system, the Kanban method of pull production, a philosophy of respect for employees and participatory management, fostering employee's problem-solving. The application of Lean thinking has spread into many other industry sectors, beyond the automotive industry (Hines, Holweg and Rich, 2004). Womack and Jones in the 1990's have promoted the lean production emulation for not-automotive and not-Japanese companies, with the aim of providing universal guidelines for managers struggling with combining lean techniques into a coherent system. *"After 1990, there was a gradual widening of focus away from the shop-floor, a trend often ignored by omission, error or design by many detractors. This process of "extension" was also accelerated by the promotion of successful western case emulation by businesses in diverse sectors that had adapted their production systems to include a new design based upon "lean principles" (Hines, Holweg and Rich, 2004, p. 995).* A crucial point in the lean thinking is the focus on values creation, moving from a merely *"shop-floor-focus"* on cost reduction and waste reduction, towards an approach that improves value to customers, by including service features and removing wasteful activities (Hines, Holweg and Rich, 2004). Value is created if waste is reduced internally, if the wasteful activities and the associated costs are decreased and if additional services are provided which are valuable for the customer. The final goal is to add customer value by a shorter delivery cycle and smaller delivery batches as well.

Womack and Jones (1996) have defined the principles of Lean, involving the identification of customer value, the management of value stream, the creation of steps flow, the pull-production mechanism and the pursue of perfection through reducing waste in production system. The first lean principle is the specification of value as stream defined by the customer, with a mechanism of decisions making based on customer expectations. Some of the following Lean practices will be considered to define the presence of an increasing creation of customer value in agile teams at Unit4. The following dimensions have been identified from Womack and Jones in 2015 in relation to customer value creation and how companies and customers can create value and wealth together.

### 2.1.2 Customers give feedback on quality and delivery performance

According to Womack and Jones, every customer is a *kaizen* opportunity, a never-ending journey towards quality and efficiency (Palmer, 2001). Companies need to train employees to explore why customers are not fully satisfied, and explore the root cause with customers and collect feedback from them, to be able to eliminate those root causes. Organizations should be able to exceed customer expectations by collecting additional information and asking about customer's needs for new services and new goods (Womack & Jones, 2015).

### 2.1.3 Customers involvement

Products and services need to work for customers in their environment, so it is important for organizations to connect with the customers; most of the companies tackle this problem with help lines, which are supposed to solve the problems of customers at a lower cost per customer, thanks to out-sourcing and off-shoring procedures. Customers might be frustrated because the direct contact with the provider of the products or services is lost (Womack & Jones, 2015). The fact of producing good services per se is not enough, because they need to work in the context of the customer, to create customer value for the ones that are purchasing them; the direct contact between customers and providers regarding the product offering is therefore necessary (Womack & Jones, 2015).

Some features of the Lean practices can be compared to agile studies, which will be presented in the next paragraph to introduce self-organizing potential. The original hypothesis in the study around critical success factors in agile software project of Chow and Cao in 2008 was *"having a strong customer involvement is a critical success factor that contributes to the successful agile software development projects in terms of Quality, Scope, Time, and Cost"* (p. 969). In their studies the creation of customer value is presented as the degree of customer involvement in the process of software development, that could be further concretized as a good customer relationship, a strong customer presence and with customer having full authority (Chow & Cao, 2008). Working on having a good customer relationship, a strong customer presence in the process of software development, and giving authority to the customers will benefit both the organization and the customers, because the organization will be able to produce software that are more valuable to the customers.

In order to survive and prosper, organizations need to involve customers to be able to understand what customers want to receive as a value; organizations need to fulfill the return on investment that the customers have done. Companies need to develop measures of performance

that capture if this purpose is being achieved, and provide the customers with what is needed, exactly when is needed. Organizations need to measure the gap between acceptable and unacceptable performance. Customers sharing of current and future demand is essential and the trick from the company side is being able to record and act upon the gap between the way it is supposed to be and the way it actually works for the customers. A continuity in sharing this type of information from the customers side is useful for rethinking the process and create customer value (Womack & Jones, 2015).

## 2.2 Self-organizing potential within Agile

Self-organizing potential has been defined as one of the critical success factors of Agile methodology by Chow & Cao in their study around critical success factors in Agile literature; a coherent, self-organizing teamwork has been proven to be one of the relevant success factor in Agile (Chow & Cao 2008). Moe, Dingsøyr and Dybå have addressed self-organizing capacity as an improvement for problem solving and team effectiveness, when a high level of autonomy is brought into the team (2008).

Agile methodology is the preferred theory to define the self-organizing potential, because this concept is well developed in agile literature and self-organizing potential is embedded in many practices and methodology that belong to the agile methodology umbrella. In this first paragraph, the self-organizing potential will be addressed at team level and accordingly the agile methodologies implemented at Unit4, such as Scrum.

To shortly introduce agile, it is important to mention that the Agile Manifesto was founded on February 2001, when the Agile Software Development Alliance was established. Seventeen recognized software developers, worked together on the creation of the agile manifesto, which was drafted with the below purpose.

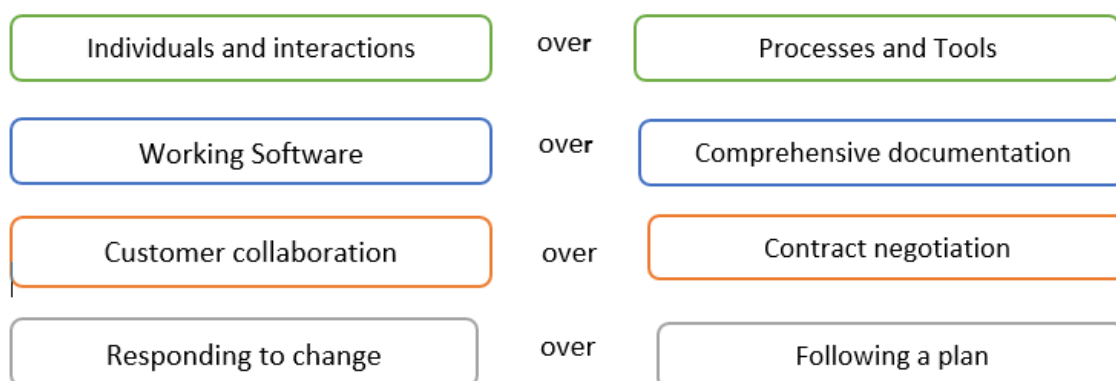


Figure 2: Four values of the Agile Manifesto (2001)

"We are uncovering better ways of developing software by doing it and helping others do it." (Fowler & Highsmith 2001, p.2). Two interesting terms have been chosen while drafting the manifesto. First the word "*uncovering*" was explicitly selected to reinsure the readers that the Alliance members do not have all the answers; second "*by doing*" implies that the seventeen participants of the Agile Alliance actually apply these methods in their own work (Fowler & Highsmith, 2001). Twelve principles have been stated in the agile manifesto; the most relevant to better understanding the degree of self-organizing potential in agile is the eleventh principle which states that self-organizing teams produce the best solutions. People in the teams are responsible for the way work is conducted, rather than being directed from someone outside the team (Fowler & Highsmith 2001). This principle will be further elaborated on in the next paragraph.

### 2.2.1 Agile team features in software development

Software development is an interesting industry to investigate self-organizing potential in teams; literature indicates that there is a significant difference between functional teams and agile teams in software development. Figure 3 shows characteristics of functional teams and agile teams.

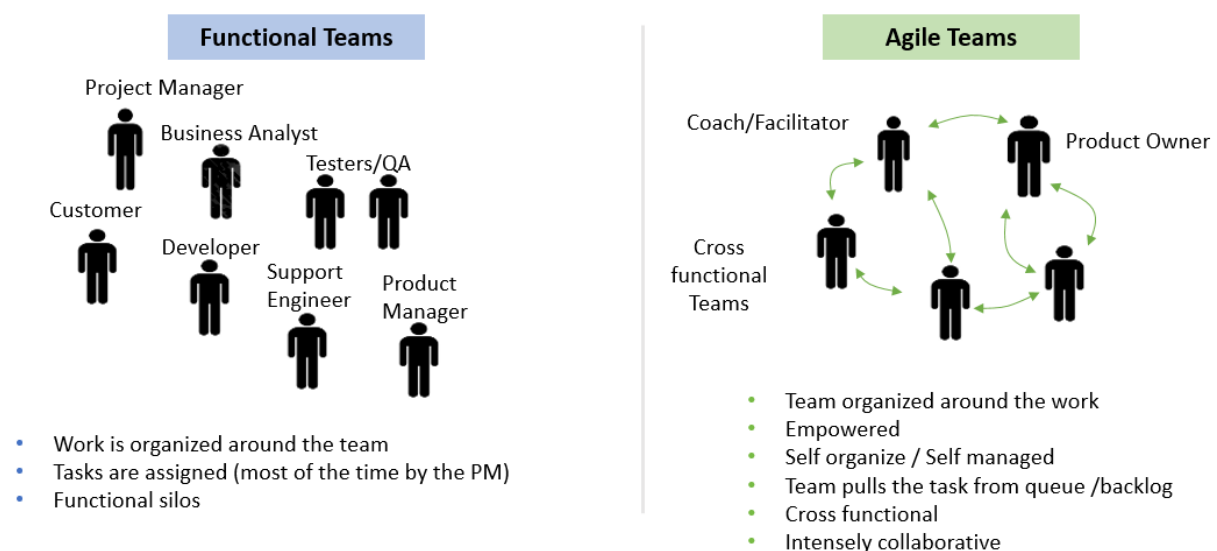


Figure 3: Functional Teams vs Agile Teams (Schwaber & Sutherland, 2016).

Being a self-organizing agile team means that the people in the teams are responsible for the way work is conducted, rather than being directed from someone outside the team. Being a cross-functional team means that inside the team all the skills and expertise necessary to accomplish the work are present within the team, and there is no external dependency; the team

as a whole is fully equipped to execute the job, counting on the internal resources and skills of the team members.

The label self-organizing teams can be used as a synonym for autonomous and empowered teams, which stimulate involvement and overall engagement with the organization. Agile development teams do not put emphasis on up-front plans and rigid plan-based control ones, but rather focus on mechanism for change management and tend to establish informal communication and organic, participative and cooperative *modus operandi* (Moe, Dingsøyr, Dybå, 2008). Self-organizing teams have been defined in the 1990's as teams of employees who are given significant authority, who are responsible for a large part of their job decisions and their economic repercussions; those teams are performing as one social unit in an organization and conduct highly related and interdependent jobs (Guzzo & Dickson, 1996).

### 2.2.2 Degree of autonomy

In 1986 the three main characteristics of self-organizing teams have been delineated as autonomy, cross-fertilization and self-transcendence (Takeuchi & Nonaka 1986). Their three main concepts do not strictly belong to the field of agile literature but could be considered the precursors of it, because those concepts are largely discussed in agile as well. Those are not included in the theoretical framework as independent variables but are necessary. Hence, they are described in this section to better understand the origins of self-organizing characteristics. Autonomy was considered as the degree to which senior management was ensuring freedom and minimum degree of interference with the team actions and decisions; the ownership of that given freedom by the self-organizing teams was equally important. *"On a day-to-day basis, top management seldom intervenes; the team is free to set its own direction. In a way, top management acts as a venture capitalist. Or as one executive said, "We open up our purse but keep our mouth closed"* (Takeuchi & Nonaka 1986, p. 139). The concept of cross-fertilization means a variation of functional specializations through behavior patterns, process in the carrying out the new product development. Bringing all the team members in one large room, facilitates the exchange of information and the interaction: *"When all the team members are located in one large room, someone's information becomes yours, without even trying"* (Takeuchi & Nonaka 1986, p. 140). Self-transcendence means that teams establish their own goals and self-evaluate and self-assess them, through their own development progress. The concept of autonomy has been largely discussed by Takeuchi and Nonaka, and it is an important aspect to determine agile self-organizing potential in software development.



In agile software development, the degree of self-organizing teams' autonomy has been further defined on different levels; external autonomy in relation to a self-organizing team with respect to the rest of the organization; internal autonomy as internal organization of the work in the group; and individual autonomy related to how individuals organize their own personal work (Moe, Dingsøyr, Dybå, 2008). *"In the following, we define autonomy as the degree to which the task provides substantial freedom, independence, and discretion in scheduling the work and in determining the procedures to be used in carrying it out"* (Moe, Dingsøyr, Dybå 2008, p.78). The degree of autonomy is defined as the degree for the team to have authority to set its own goals (goal-defining autonomy), to define its own structural autonomy, based on social identity and social system boundaries (structural autonomy); to the authority to define the behaviors of the team member (social autonomy) and freedom to choose the resources required to accomplish the self-assigned tasks (resource autonomy) (Moe, Dingsøyr, Dybå 2008).

### 2.2.3 Scrum framework – degree of Scrum implementation

Scrum is part of the agile movement as an agile method, which is currently implemented at Unit4 in the R&D department. Scrum method is an iterative incremental process of Software development, that can be used to control and to manage software and product development by using incremental practices (Hu, Yuan, Zhang 2009).

Scrum is defined as a management framework in which people can address complex problems, while productively and creatively delivering products of the highest possible value (Schwaber & Sutherland, 2016). The Scrum framework provides a structure of roles, rules, artifacts and meetings that team members need to follow in order to achieve their goals. Scrum is based on empiricism, especially, on the empirical process control theory. Empiricism asserts that knowledge derives from experience and decision making is based on what is known (Schwaber & Sutherland 2016).

In order to work in an agile way, teams must go through a Scrum transformation and embrace the three main pillars of scrum which are: transparency, inspection and adaptation.

*Transparency* means that all the relevant aspects of the process must be visible to those responsible for the outcome. Those aspects should be specified by a common standard, and a shared common understanding. (A simple example of transparency is the definition of a common language and the sharing of the common definition of "Done", for the people involved in the output delivery. *Inspection* means that scrum artifacts must be frequently inspected. The inspections should not stop the regular performance of the job, but should aim at detecting the

undesirable variances. The defined aspects of the process should be respected. *Adaptation* means that if some variances are identified in the inspections, the scrum teams should make an immediate adjustment, in order to minimize further deviations.

Scrum is an iterative and incremental approach to optimize control risk and predictability, and maximize feedback; it uses fixed-length iterations, called Sprints, which are no more than 30 days long. In every Sprint, agile teams should build a potentially releasable product, which has been properly tested. Scrum teams have a particular team composition, which consists of Product owner, Development team and Scrum Master. Those roles are defined in appendix 8.5 and below figure 4 shows the Scrum methodology for a better understanding of the roles and processes.

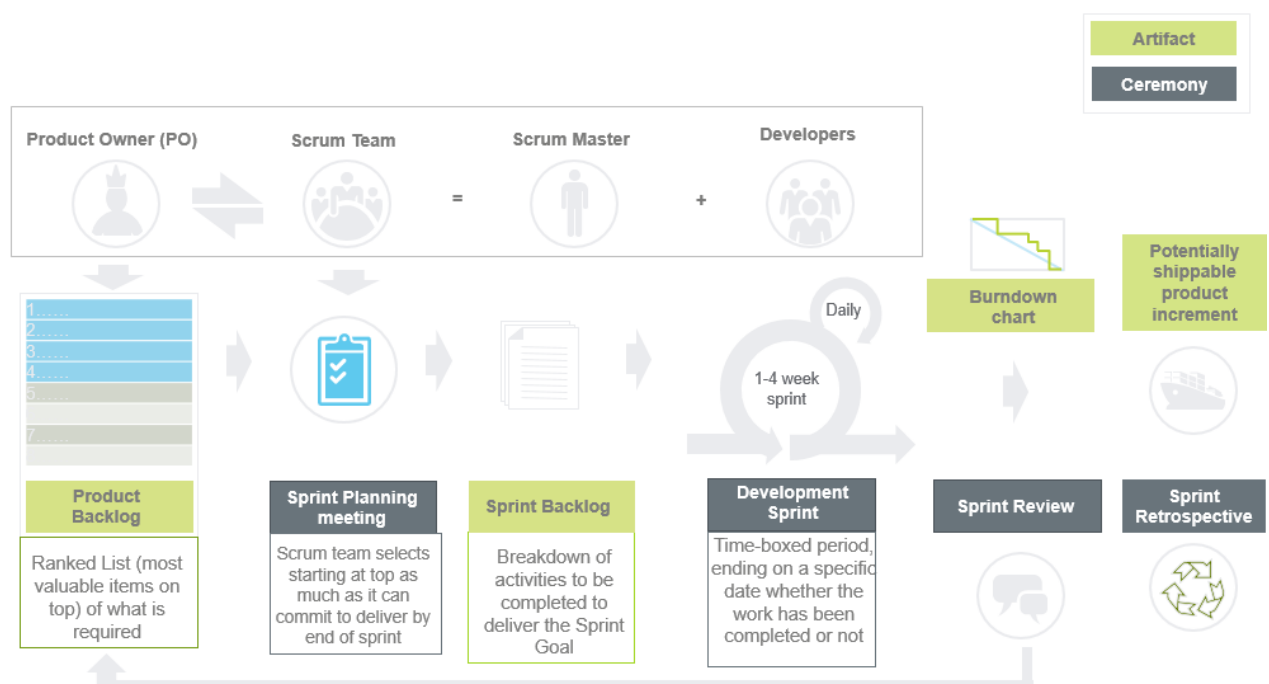


Figure 4 – Scrum Methodology (Schwaber & Sutherland, 2016).

## 2.3 Structural characteristics – Sociotechnical Design theory

As was highlighted in the introduction, this thesis aims to study an integral perspective on organizational design; in the following paragraph the Sociotechnical theory will be explained and its possible accelerating effect on the existing relation of self-organizing potential and customer value. Sociotechnical design theory is a Dutch approach to organizational design which focuses on the integral redesign of organizations by looking at the main structural and architectural parameters that define the production and control structure (De Sitter et al., 1997). The structural

characteristics are formulated based on the idea of reducing organizational complexity, which is the primary concern of the Sociotechnical design theory.

In relation to this case study at Unit4 the structural characteristics of Sociotechnical design theory will be investigated as possible accelerators of the relationship between the self-organizing potential and customer value in agile teams. While self-organization is a crucial characteristic of Agile, there has been limited research on the subject and almost none across multiple projects, organizations, and cultures. For example, as previously mentioned, Moe et al. (2008) note that Scrum explains self-organizing teams but does not provide directions regarding its implementation and how the organizational structure should be adjusted accordingly.

The following statement is the reason why an accelerating influence of the relationship between self-organizing potential and customer value is hypothesized: *“Sociotechnical theory explains how a specific architecture determines the opportunities for coordination, adaptation, and innovation of system-internal and external functions. Sociotechnical design is concerned with creating and using such opportunities by changing the architecture.”* (De Sitter et al 1997, p. 506). In this thesis, the relationship between internal self-organizing potential and external customer value is hypothesized to be accelerated by a more integral design approach at Unit4. Therefore, the relation between self-organizing potential which is system-internal, and the creation of customer value which is external, can be hypothesized to be accelerated by including structural characteristics of De Sitter into the conceptual model of this thesis.

In the next paragraph the structural characteristics will be explained and the mechanism by which those are proposed to accelerate the relationship between self-organizing potential and customer values is explained. The following main structural characteristics of the production structure have been selected to address the organizational design component in this case study to prevent a too complicated conceptual model, and with supporting arguments from theory as the most important parameters: functional concentration and level of differentiation of operational transformation. De Sitter distinguishes between the concepts production and control structure, but for this study the researcher has selected two parameters from the production structure for the reasons expressed when presenting the structural characteristics below.

### 2.3.1 Functional concentration

The first structural characteristic taken into account in this theoretical framework is functional concentration; this choice has been made because this has been defined by De Sitter as the most important one in comparisons to the remaining parameters; *“This structural*

*parameter is perhaps the most important one because high functional concentration limits very much the freedom of choice with respect to the remaining parameters and is responsible for deficiencies with respect to delivery time, quality, marketing, quality of working life, innovative capacity, etc. Functional concentration is still a dominant feature of most current production systems.” (De Sitter et al, 1997, p. 507).*

This first design parameter, the degree of functional concentration, is related to the production structure and has been introduced by De Sitter, *“referring to the grouping of operations (operational task) with respects to orders”* (Achterbergh & Vriens, 2009, p. 243). A maximum value of functional concentration indicates that all operational tasks of the same type are concentrated into specialized departments; while a minimum level of functional concentration means that all operational tasks of different type are grouped into a “production flow” (Achterbergh & Vriens, 2009, p. 243).

In a structure with maximum level of functional concentration employees execute tasks related to all possible order types. Below, figure 5 shows an example of high functional concentration in the production of tables and chairs, in which all the operational sub-transformations (such as sales, planning maintenance, sawing, drilling etc.) are related to both tables and chairs production (Achterbergh & Vriens, 2009).

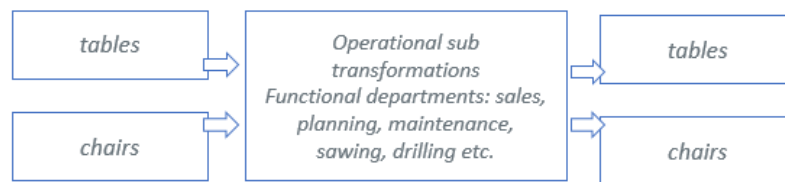


Figure 5: Maximal degree of functional concentration, Achterbergh & Vriens, 2009, p. 245

In a structure with minimal level of functional concentration the employees execute tasks only related to one order type; in this type of structure there might be “parallel flows”, coupled to order type. See an example of functional concentration in figure 6 below, where two parallel flows are drawn (one for tables and one for chairs) (Achterbergh & Vriens, 2009).

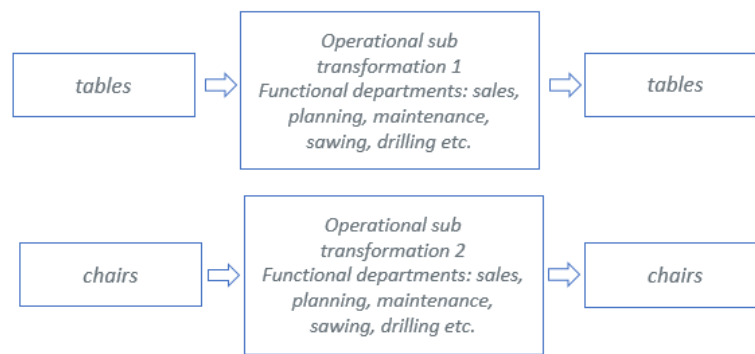


Figure 6: Minimal degree of functional concentration, Achterbergh & Vriens, 2009, p. 245

### 2.3.2 The level of differentiation of operational transformations

The second parameter is important because it helps defining the three types of operational sub-transformations that have been differentiated according to De Sitter as “*making*”, “*preparing*” and “*supporting*” Achterbergh & Vriens, 2009, p. 246). *Making* refers to all the activities directly related to the realization of the output; *Preparing* includes all the activities necessary to start making the output, providing the conditions to perform the sequence of activities. Both making and preparing are linked to the specific output transformation, while *Supporting* includes all the activities that support the output realization, such as finance, human resource’s planning and technical services (Achterbergh & Vriens, 2009, p. 246). The second parameter describes the level of differentiation of operational transformation. It is at an ideal minimum level if making, preparing and supporting activities are contained in operational sub-transformations. If the activities are specialized and grouped per category, the level of differentiation is at its maximum level (De Sitter et. Al, 1997).

Sociotechnical design theory has been defined as a theory that focuses on work organizational structures, which aim at reducing organizational complexity (Christis & Soepenbergh, 2015). With regards to organizations, that means that complexity increases with the number of interfaces, the amount of communications needed between interfaces and the variability of information that flows through each interface. De Sitter et al. (1997) argue that increased complexity will increase the probability of disturbances and decrease the potential for regulation within a system. In this case many disturbances can be linked to the organizational structure of an organization; both functional concentration and the level of differentiation of operational transformation can play a role in reducing disturbances and preventing them. The

following solution in fact is suggested by De Sitter et al. (1997). An organization should ensure that the potential of those disturbances is as low as possible, by reducing functional concentration and increasing the integration of various tasks. There will be less variability and less relations that the employees will need to deal with on a daily basis and this will consequentially reduce the probability of errors to arise. Each job in the organization will become more fulfilling and challenging for the employees and teams themselves.

## 2.4 Relations between concepts

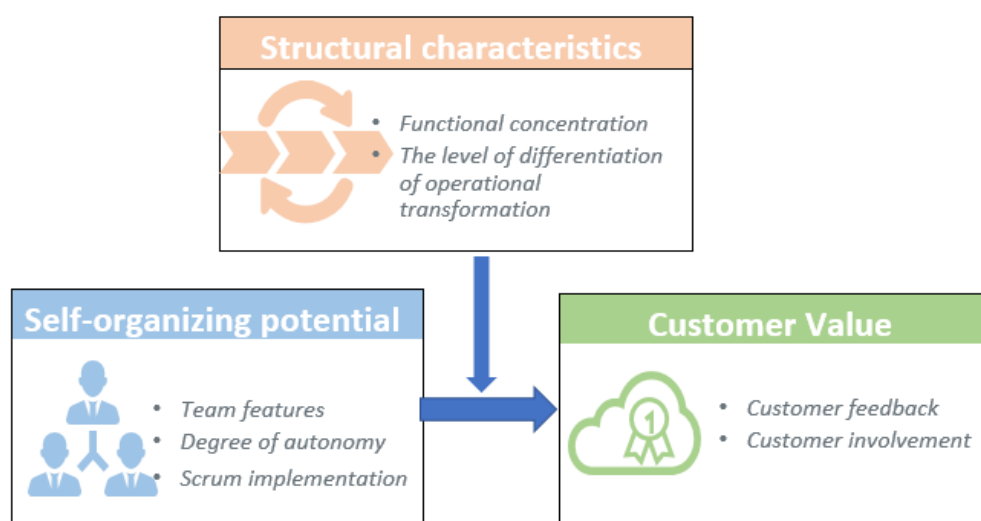


Figure 7: Relations between main concepts and theoretical framework.

The conceptual model, in figure 7, summarizes the relations between the main concepts and the theoretical perspective to which they belong in the theoretical framework, keeping in mind the context of agile teams and software industry. Creating additional customer value is one of the main goals of Unit4, and the self-organizing potential, addressed in an agile perspective, should result in a strong customer involvement. From a Scrum methodology perspective, for example, the customer is involved by the self-organizing team, at the end of each sprint.

The self-organizing potential will be addressed from the Agile perspective; the creation of customer value will be addressed from a Lean and Agile perspective; and the structural characteristics and the mechanisms will be addressed from a Sociotechnical design theory as possible accelerators of the relationship.

### 3 Methodology

After having introduced the topic, its relevance and the main concepts, this chapter will clarify the methodology used in the study. By doing so the general case study approach will be clarified and the data collection methods chosen. Finally, ethical considerations will also be included in the following chapter.

This master thesis research is conducted with qualitative methods, and the nature of the research questions is exploratory and in a fixed point in time, nevertheless recommendations on the implementation of agile could be drafted in the conclusive part of the thesis. The research is done by a qualitative case study at Unit4 and data will be collected via interviews. Qualitative research is a desirable approach for understanding a phenomenon while taking into account the context in which the phenomenon is studied (Justensen & Mik-Meyer, 2012). Furthermore, qualitative research is a meaningful method in early stages of exploratory research, for theory building and understanding a phenomenon, while capturing the richness of the experience (Yin, 2017).

#### 3.1 Case study and systematic combining

A case study has been defined as *“an empirical inquiry that investigates a contemporary phenomenon in depth and within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident”* (Yin, 2017, p. 13). This method will be used to gain in-depth insight into this specific organizational situation at Unit4, to gain information in a real-life context of agile teams. This is a single-case study because it is within the context of one single software organization. Case study research usually focuses on answering ‘how’ and ‘why’ questions, such as the given research question and the two sub-questions (Yin, 2017). The case study depends on a selective sampling of a relevant case, from which learnings can be gained (Yin 2017), such as representatives of agile teams that have been self-organizing their works and that interact with customers.

Data collection and theory will be combined, in an iterative process with the aim to contribute to the existing literature, with a mix of a deductive and an inductive approach (Dubois and Gadde, 2002). A systematic combining approach suggested by Dubois and Gadde in 2002, is a mix of inductive and deductive approach, in which theory cannot be understood without empirical observation and vice versa. This is a more suitable approach than the pure grounded

theory methodology for this case study because it gives to the researcher more flexibility during the data collection process and analysis; according to the systematic combining approach a fixed number of subsequent phases does not fulfill the potential of the case research, because most of researchers constantly go back and forth between observations and theoretical framework, which is a positive way to understand both theory and empirical phenomena. The systematic combining approach is represented in figure 8 (Dubois and Gadde, 2002).

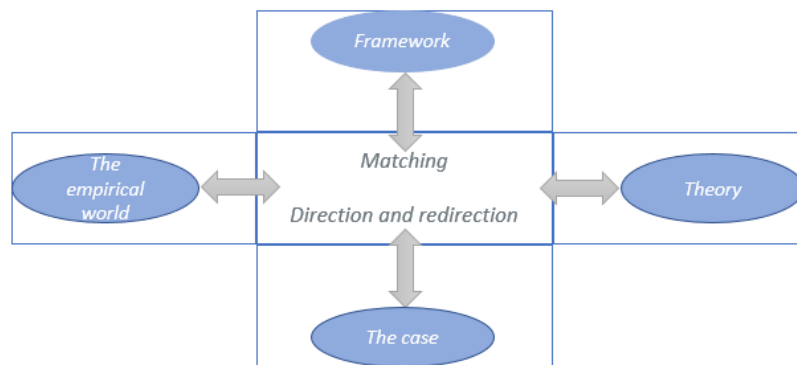


Figure 8: Systematic combining

### 3.2 Qualitative data collection – interviews and sample

The preferred method to collect data is via conducting interviews, in order to gain fast and deep insights into the situation (Yin, 2017). Exploratory interviews have been conducted mostly to gain insight into the R&D department at Unit4 and on the self-organizing potential in agile teams; the agile methodology perspective has been investigated in two exploratory interviews in February 2018. In a second phase, semi-structured interviews were conducted, to gain insight on the research question and sub-questions and to discuss and draw conclusions on the research. Interviews are a conversation with a purpose, driven by research interest but open and flexible (Yin, 2017). They are a suitable method to answer to “how” and “why” questions. The collected data is interpreted and not taken for granted, in the analysis and coding procedure; the main concepts are operationalized once the data was collected. Documents, such as, the company website, some blog articles, the organizational structure and other informative material are also considered.

The sample of the interviews has been chosen based on the contacts provided by the manager of Benelux R&D Domestic and by the Quality Manager of the R&D who are also the sponsors of the Agile transformation at Unit4, based on the assumption of an appropriate level



of collaboration with customers. The interaction of those teams with customers have been analyzed during the interviews and it makes it sufficient, considering the scope of this Master thesis, to assess the customer value creation from Unit4 perspective, based on the feedback and the involvement of the customers. Four interviews have been performed within four R&D teams, that have been selected from two lines of business: Financial Services and SME. Three follow-up interviews have been performed with the same teams, interviewing more people with distinct roles belonging to the same team. For example, by interviewing both the Product Owner and the developer of the team, it was possible to take different perspective into account and a full picture of the situation. This decision has been made to gain in-depth and saturated understanding of the teams, rather than enlarging the sample size to other lines of business. This choice could raise some issue in relation to internal validity of the research and is a limit of this master thesis. Elaboration of validity and reliability will be mentioned explicitly in the Discussion related to the limitations of this master thesis.

In the results section the researcher will be referring to the four teams based on their types of activity performed. The code of the transcript will facilitate the researcher in allocating the employees in the right team, guaranteeing anonymity for the participants of the interviews.

<b>Team 1:</b> maintenance and new development activities.	<b>Team 2 and Team 3:</b> new development activities only.	<b>Team 4:</b> front-end team, not yet on the market.
01	02	07
05	06	04
	03	

*Table 1: Teams and types of activities performed.*

### 3.3 Operationalizing and Analysis

In qualitative research, operationalizing is an iterative process (Blackstone, 2016). In the theoretical framework the main theories involved in the thesis have been specified in their key concepts and dimensions. An example of a semi-structured interview guide can be found in appendix 8.2, with examples of questions defined per each dimension. In appendix 8.2 the main concepts have been summarized in a table, with related indicators and examples of possible questions.

The interviews have been conducted with the permissions of the interviewees, and typed into transcripts. The coding process has been conducted, both deductively and inductively, with

stages of open, axial and selective coding. Systematic combining can be described as a nonlinear process of combining efforts with the ultimate goal of matching theories and real world (Dubois and Gadde, 2002). Blumer (1954) suggests that theoretical concepts should be used in a sensible way in order to create a reference and to function as a guideline when entering the empirical world. Similarly, Bryman (1995) states that a theoretical concept provides the researcher with a set of general guidelines when analyzing the results (Dubois and Gadde, 2002). Keeping in mind that in a systematic combining perspective, the researcher's objective is to discover new things, but systematic combining builds more on refinement of existing theories than on inventing new ones (Dubois and Gadde, 2002).

Coding in the open and axial phase is conducted deductively and inductively, in the sense that the results emerged from the interviews inductively, but under the guidance of the concepts presented in chapter two. In a more selective phase the link with theories and the conceptual framework is done explicitly. A sample of coding statement cards has been provided in appendix 8.4, together with the axial map, representing the axial categories of paragraph 4.1 (axial result).

### 3.4 Ethical considerations

The research landscape is changing and has been accompanied by a rapid increase in research ethics regulation and governance. The use of data and the interpretative and analysis process have all become significant as the landscape of qualitative research is changing and researchers produce knowledge (Miller, Birch, Mauthner & Jessop, 2012).

This research is conducted respecting ethical procedures. Personal information of the participants and Unit4 is treated confidentially and anonymity is assured, for example by storing names and contacts separately from the transcripts and consistently using codes instead of employees' names (Hammersley & Traianou, 2012).

At Radboud University ethics within a research is a very important topic. During the master program, ethical considerations have been treated in the research course and the following points will be elaborated and taken into account. Anonymity is granted for the organizational members who have participated in the interviews. Personal information of the participants and Unit4 is treated confidentially and anonymity is assured, for instance by keeping names and contact details separately from the transcripts and consistently using codes instead of names (Hammersley & Traianou, 2012).

Unit4 will have to approve a publication of the thesis, otherwise the thesis will be available in anonymous form, without mentioning the name of the company. A copy of the thesis will be made available to the organization and to the people who directly contributed to the interview process.

Participation in the interviews was gained on a voluntary basis, and after the signing of the informed consent form (See Appendix 8.3). The informed consent form has been provided to all the participants in the interview process and it has been signed by all the participants.

## 4 Results

What follows is a description of the results and of the main findings, which emerged from the interviews conducted with the Scrum teams in the R&D Benelux department.

In the 4.1 result section, an axial phase will be reported. This choice is made to express via the employees' own words the main categories which emerged with a systematic combining approach, by keeping the concepts in mind. This will give more insight into the interviews and how the analysis was made.

Self-organizing potential
Degree of autonomy
Team features
Degree of Scrum implementation
Customer value
Customer Feedback
Customer Involvement
Structural Characteristics
Functional concentration
The level of differentiation of operational transformation

Table 2: main concepts used for conducting semi-structured interviews

The paragraph 4.2 is an analysis of the results with an explicit link to the theories; the main findings related to the self-organizing potential and the creation of customer value, considering the team features, the degree of autonomy, the degree of scrum implementation which are contributing to the value creation process (sub-question 1) and the moderator effect of the structural characteristics considered in the theoretical framework (sub-question 2).

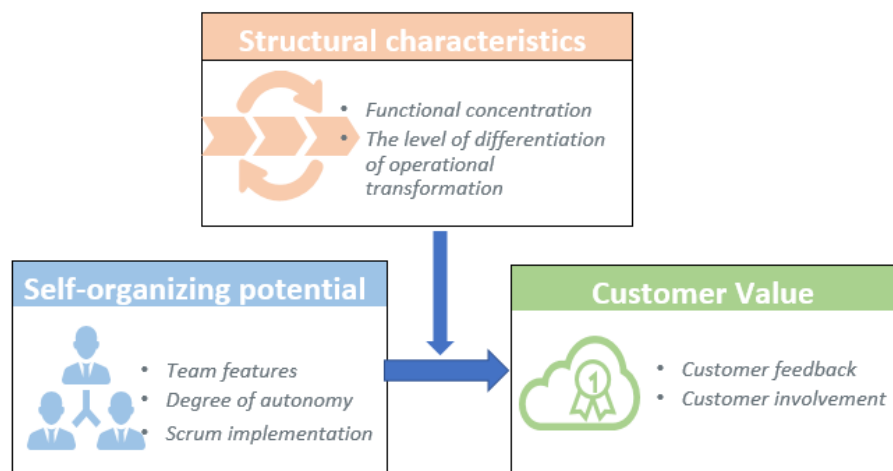


Figure 7: Relations between main concepts and theoretical framework (chapter 2 Theoretical framework).

## 4.1 Axial phase

In this axial phase three main categories will be presented, which will give a first indication of the concepts:

- *Working with a Continuous Delivery model*
- *Dealing with the DTAP model*
- *R&D working with other departments*

Those categories are the Unit4 findings based on the employee's answers to the semi-structured interviews built on the concepts of this thesis. Please consider that the continuous delivery model and the DTAP model are not concepts from the theoretical framework, but are Unit4 models that employees use and work with when they apply theories in their work. The concepts are underlying in those Unit4 models; the relations between concepts is made more explicit in the analysis part. The results of working on self-organizing potential to create customer value is what enables teams to work with those models, which are Unit4 ways of working following Scrum principles and other concepts presented in the theoretical framework.

For example, working with a continuous delivery model reflects the adoption of Scrum principles in the developers' job at Unit4, which is a strong component of their self-organizing potential to implement customer value. In the next paragraph more findings will be presented and supported by quotes from the interviews.

### 4.1.1 Working with a Continuous Delivery model

What seems to be a common result from the interviews, is the recognition of a so-called continuous delivery model for achieving customer value. In this section it will be described how teams work on the creation of customer value by using this method and insight into the concepts presented in the theoretical frame will be given. It will be shown that by working with this continuous delivery model, employees work on self-organizing potential, and achieve customer value.

#### 4.1.1.1 *Improving self-organizing potential and customer value while working with continuous delivery model*

Teams work with Scrum principles, which seems to improve the creation of customer value. The consequence of working following Scrum principles results in the practice of Unit4 continuous delivery model which creates value for customers. While improving the self-organizing potential Unit4 teams follow the so-called continuous delivery model to release valuable software to the

customers. Most of the teams work with iterative Sprints, which is part of their continuous delivery method: *“we adapt to what is necessary, so the priority can change from now and then, but also you can adapt based on the feedback from the customers, we can implement and change what we are creating from the next 3 weeks”* (06). At the end of the iterations it is important to deliver software that is valuable for the customers, by adapting the products and the deliverable priorities, based on their feedback. The link with theory will be made more explicit in the analysis paragraph (4.2).

It is relevant to report that the results include interviews with employees working on very mature software, dealing with legacy issues and maintenance (Team 1), and on very new products on the market (Team 2 and Team 3); but also, an interview within a team that is working on new product which is not yet on the market (Team 4) was conducted. This difference between teams is relevant because it impacts how teams can improve self-organizing potential and create customer values, while working with a continuous delivery model. (See table 1 in the methods section). At Unit4 the continuous delivery model is applicable for the teams that have their product on the market, including both maintenance and new development (Team 1; Team 2 and Team 3) and is not applicable for Team 4 for which it is not on the market yet.

Team 2 and Team 3, seem to have all the team characteristics of agile teams, seem to work autonomously and seem to have implemented Scrum principles at a very good level in their practices; this results for them in delivering continuously to the customers in practice, following Sprint releases of their software. The fact that they are working on their self-organizing potential enables them to work with a continuous delivery model. Working with continuous delivery has an impact on the creation of customer value, and there are significance differences in the value created for customers in teams that use this Unit4 delivery model constantly and teams that struggle with it.

Respondents point out that working with continuous delivery model is sometimes easier or harder depending on the type of activity that they perform, and this difference between types of activity is impacting how teams can improve self-organizing potential and create customer values.

Team 2 and Team 3 which are working on new development do not need to deal with maintenance activities as Team 1 does. The maintenance activity of Team 1 can be time consuming and take away focus from their self-organizing potential and the delivery of new features or of new user stories at the end of each Sprint. Maintenance mainly consists of solving

bugs. The reasons and the implications of the differences between those teams will be explained below in this section and are visually represented in figure 9 and figure 10 in the next page.

One of the reasons why the team working with more maintenance activities (Team 1) is still partially holding on to the more traditional way of creating software is due to legacy reasons from the past and to the fact that the traditional way of releasing is a profitable business model.

*“Once in 10 months we release to between to 400-500 companies that use our software and by releasing our software we earn money by installing a new version. When a customer pays the installation consultant for a new version the services cost 6 thousand euro. If we stop that model and we go to a continuous delivery model, part of that income will fade”. [...] We start in month 1, we make things and we put it on the shelf, and month 10 we release it and we get feedback of our customer. So that’s not really working agile.” (01).*

In the 10 months-release models, it is not possible to implement all the feedback received by the customers, without throwing away the software and building it again from scratch, which is a waste of time. So, it is possible to conclude that holding on to a more traditional way of working makes it harder for the team to involve the customers and to collect and to implement feedback from the customer; this diminishes the creation of customer value for the customers. This is presented in figure 10 below.

Teams that are fully working with Unit4 continuous delivery model seem to have implicitly embraced scrum concepts and seem to have developed an appropriate level of self-organizing potential. In contrast to the more traditional way of 10 months-release described before, is the continuous delivery way. All the four teams are at least in the pilot phase with the continuous delivery. Some teams seem to have embraced Scrum principles completely (Team 2 and Team 3) and are fully working with a continuous delivery model:

*“What we are now doing is that we want to release the software as often as possible; for us this is 3 weeks now for the pilot. But by doing that you also want to ensure quality” (02).*

Team 2 and Team 3 have embraced Scrum principles in a complete way and they work easier with a continuous delivery model, producing customer value for their customers, as is shown in figure 9 below.

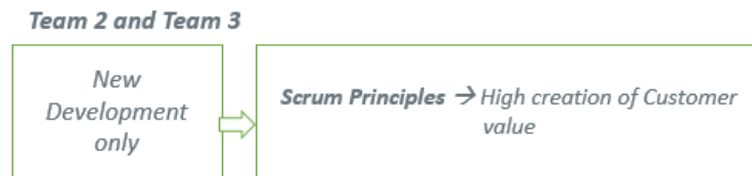


Figure 9: Team 2 and Team 3 implemented Scrum principles and are continuously delivering to the customers

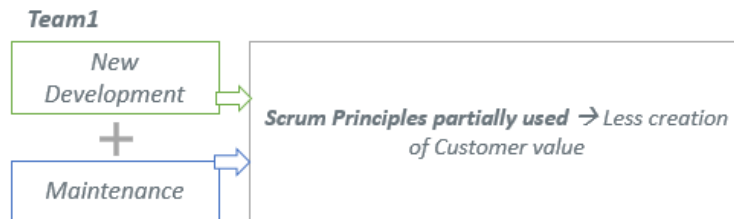


Figure 10: Team 1 partially implemented Scrum principles and it is working with continuous delivery and with a more traditional way of delivery

While working with this delivery method, teams give a lot of importance to the customer involvement and collecting customer feedback which both is always in the mind of the Product Owners and the developers. There are several ways to ensure customer participation in the continuous delivery process: via the creation of user's community, creating interaction with the customers and the developers directly: *"That is where the business value comes from in creating a community. And it is a nice way to have the real end users and the developer talk to each other. For developer it is scary to have a customer on the floor"* (02). At Unit4 this is happening with a platform: *"There is also a platform called "Unit4ideas" so customers can add an idea or wish and other customer can vote. And when there are enough votes we can implement that idea."* (05). At Unit4 the use of a co-creation (pilot) with customers, the Unit4ideas platform, and the collection of feedback before developing are highly considered by developers because: *"the customers feel that they have some influence on the products, and just get it out of their systems and they would know that who is behind the products has a face; they would get more emotionally attached to the company and to the product. Those are big advantages for the customers. They don't feel treated as a number, but they are heard and they feel free to say everything to improve our product"* (07).

At Unit4 it is not yet established to involve the customers at the end of each Sprint for feedback, even though the Product Owners of most of the teams are currently working on making this happen by the end of the year, to ensure a direct contact with the customer and shorten the feedback loop: *"The shorter the feedback loop gets the better we can work on the actual problem, instead to work on something that is translated by Sales and by Marketing for example"* (06). This



will be discussed in the 4.1.3 paragraph, where the struggle of R&D in dealing with other support departments will be explained.

#### 4.1.1.2 *The importance of autonomy, while working with a continuous delivery model*

All the teams recognized the importance of working in an autonomous way while working with a continuous delivery model; it is important to be able to plan own work and the team work independently to deliver highly valuable products for the customers. Working with this so-called continuous delivery model also means working autonomously.

If the Sprint planning and process is done autonomously, the Product Owner has more time to engage with a customer's relationship, to get more feedback and to engage the Development Team in a better relationship with the customer. The Development Team needs to think independently and take own responsibilities for their actions and for the products that they develop; ideally the Product Owner sets the goals and the Development Team confirms if the goal is achievable demonstrating independence in the decision making-process, within the Sprint (two weeks/three weeks/four weeks, the time frame can variate from team to team).

*"I only tell the team What we need and Why. What is needed and why is needed. [...] How is the other question and it is for the people making the software; How it's up to the Development Team. They are the specialist" (02).*

If the Development Team works independently, the Product Owners have more time dedicated to the customers and to plan and organize more meetings with the customers and the Development Team, which means that the focus on the customer is higher. Working autonomously would facilitate the connections between developers and customers, which in this moment is quite rare. The developers need to have a direct contact with the customers, and some of them have it, but mostly due to their personal background as consultants, or due to their lifelong experience with Unit4.

*"The biggest advantage is the direct line because there is no communication loss, with the consultant or product manager, or account manager or partner" (05).*

This is a powerful statement which underlines the involvement of a direct line of communication between developers and customers, without losing part of the customer's feedback in the transfer of the information. At this moment the Product Owner is the one who is most in touch with the customer and the direct contact between developers and customers is rare. This means that the Product Owner translates the wishes, desires and feedback from the customers to the Development Team, by setting and prioritizing the Sprint goal, in consultation

with the rest of the team. There is still a lot of indirect messaging, because usually the feedback passes via Sales or Consultants or Help desk, and this causes some loss of information. This will be addressed in paragraph 4.1.3 when elaborating on R&D working with other departments.

Another point of attention is the involvement of the customers with the Development Team which has a double effect. First, as mentioned before, it is valuable for the customers to implement their feedback in a shorter time frame. Second, it facilitates the customers in understanding the point of view of the developers, it facilitates the creation of a mutual understanding: *“People tend to have no idea on how development looks like, or how much work that is to do a small thing, what kind of problems you get. I think that during one of these reviews, to a smaller or larger extend, nobody could completely understand what was going on. But that was okay because they could see that we were seriously trying to make something good.”* (03). The involvement of the customers in the reviews adds a *“look into the kitchen”* for them and a better understanding of the development process. The message here is that autonomous teams have higher chances to involve the customer and create moments of exchange of precious feedback between customers and developers, with the final goal to maximize customer value.

#### 4.1.1.3 *The key role of the communication and collaboration*

Communication and collaboration are essential components to be able to deliver high and valuable products to the customers, achieving the Sprint goal. The common result of the interviews is the emphasis of a very good internal communication and collaboration within the team, which allows the team to work independently and to be autonomous with their working activities. This enables teams to continuously deliver to their customers. As mentioned before *“How”* to achieve the Sprint goal is up to the team and this requires interaction and collaboration between the developers, testers and analysts. While the internal communication is good, the external communication with other R&D teams and with other departments could be improved, in order to not lose any information and create higher customer value. The developers recognize that working Agile has improved their communication skills and the set-up of the Sprints (with the *Dailys* and the *Sprint reviews* for example) forces everyone to communicate and interact with the group.

*“I think every programmer would just like to sit in front of a screen, type things and make programs and nothing else. [...] I think that 2-weeks iteration helps keep people involved with each other and more involved with the project”* (03).

In this way people are more committed and can achieve a better result, and develop a better software.

#### 4.1.2 Dealing with the Development, Testing, Acceptance and Production (DTAP) model

In this section it will be described how by using this deployment method, teams work on the creation of customer value and some first insights into the concepts described in the theoretical frame will be given. The DTAP is part of a continuous delivery model and the advantage of the complete implementation of Scrum principles of section 4.1.1 is valid for this category as well.

##### 4.1.2.1 *Multi-skilled and autonomous team to create customer value, while working with DTAP process*

The DTAP process increases value for the customers by ensuring quality of the product and it is part of the continuous delivery model and it has a very good impact on the quality insurance of the products for the customers. Working with a DTAP model is a direct reflection of following the Scrum principles of *Inspection* and *Adaptation*, because it is an answer of Unit4 teams on how they inspect their software, without stopping the regular performance process.

Multiple skills are needed, such as developers, testers and analysts, within Scrum teams, to be able to complete the process and ensure a highly valuable product for the customer to test in a stage or production environment. Team 3 for example, does not have any official testers or analysts in the team and this creates more difficulties in testing the software for the customers in the testing phase, because developers need to play the role of testers. A multi skilled team instead has a positive impact on the customer value and on the creation of team shared responsibility. The deployment strategy of the DTAP is happening within the Sprint, and it stands in contrast to a more waterfall approach where a release is happening every six or ten months.

*“In this moment we are focusing on our DTAP deployment strategy. The traditional way of building software is release and this could take 6 months or 10 months. [...] What we are now doing is that we want to release the software as often as possible; for us it is every 3 weeks now for the pilot. By doing that you also want to ensure quality for the customer. The Development environment that is where the software is made, the Test environment, the Acceptance environment and the Production environment. In the 3 weeks we create all the development stuff, the tester will test that in the Test environment. Then it is placed on the Acceptance, and when it is there we have a review meeting and one of the question*

*at the review meeting should be “Do you want to have it?”. When the Development Team says yes, we push on a “button” and it goes to Production. So, contains the continuous adaptation flow” (02).*

The Development Teams need to take ownership for their own work and “accept” the additional features, epics or user stories that have been developed. In the teams that are developing a new product they mostly develop epics and features. Epics are big features, and epics are always something that customers need and request. If the epic is very small then it is called a user story, when it is a little bit bigger it is a feature and when it is even bigger it is an epic. Figure 11 in the next page depicts a visual representation of an *Epics Board* captured at Unit4 Sliedrecht office during the interview process.

In relation to the concepts this means that working with Unit4 DTAP, in their words implies multi skilled teams and autonomous teams to ensure quality to the customers. Autonomous and multi-skilled teams increase value for the customers by ensuring quality of the product, which is happening if the software developed goes through all the phases of the model: *“Finally, we are now in Stage environment, we need more and it will be Production which is ongoing [...] It’s quality for the customers at the end, if the product goes through every stage the quality is ensured and the customer should be happy with it.” (07).*

An important part of the Production phase is the customer feedback because the Production environment is linked to the end-users of the software. In this phase the real customers of the product test the product directly, which is an essential phase for the Development Team to demonstrate self-capacity to correct the errors and collect feedback from the users in real-time via the analysis of the analytics coming directly from the end-users.

*“We have a tester in the team. There are distinct stages for quality insurance; the first step is the tester in the team. So, each team has a dedicated tester to test what has been built by the developers; the next stage is acceptance and then it gets pushed to production. In production we have also a lot of analytics. For example, we just pushed something to production and we saw a lot of errors coming from the same source, that were images that the users couldn’t receive. [...] So that was something that we will add to our bug list and that we will fix probably in the upcoming Sprint.” (06).*



*“We have to be as independent as possible, if we would have to wait for another team to create something we could not commit to the customers. Otherwise we do not deliver value to our customer [...] Like I said, if one team should fix something, before we can create customer value, the other team does not create customer value. We want the team as independent as possible which also means that we have to work on the full stack of technological problems. Then one team can bring value, and the other team can bring value as well” (06).*

*“I think we are pretty independent because we can decide what to do, we can develop and we can release. So, we are not dependent on any other team” (05).*

Most of the teams are working independently on one single software package, and have their own pool of customers to satisfy, with one exception. In the result there was one exception (Team 4) in which the front-end team depends on the back-end team (and vice versa), and they would see both advantages and disadvantages of being dependent on each other.

*“In this moment we are busy to get to arrange that in the back-end we are doing the work and afterwards the front-end can pick it up. We have some extra functionalities that are ready, finished in the back-end and then afterward the front-end can implement [...] From the Sprint goal, we couldn’t finish because we had to wait the front-end or we had to wait since the back-end was ready.” (04).*

The split between the front-end and the back-end happened because Unit4 bought many small companies, with different solutions, developed in different languages, and that were hard to integrate. That is why Unit4 set up this new common back-end teams, to develop solutions for multiple products, but all written in the same language and with the same technique, to create more uniformity among Unit4 software. Both advantages and disadvantages of being dependent on each other have been explained: *“The advantage (of being independent from the back-end team) would be no dependencies. You can rely better on your own pace. So, you don’t have to wait. It’s always wait for someone to deliver and then we can go on. The advantage of your own independent team is that you go faster” (07).* The disadvantage of being independent is that together you can achieve much more, if you can reuse components, which are used for other products. Everybody has to deal with CRM (customer relationship manager), or log in or SSO (single-sign-on). Those are the things that all the Unit4 software need to have in the same way.

The only type of support activity that the developers are performing is third line support, when the expert is not able to fix a bug for example and a more technical person is necessary. The activity performed is *“the so called third line support. So, if first line and second line support [...] cannot answer the question, it comes to the developer to R&D. And that’s what I do also.”* (05).

Every team (every product) has a dedicated support team that does first and second line support. Sales and Marketing interact directly with the customers and a lot of communication is lost when reported back, which is not good for the customer. There is large feedback loop and some communication is lost in transferring the communication from the customer to the development team.

*“Our feedback loop is so large; there is a lot of indirect messaging, because what customers want has to go via Sales to someone here on Product management side, and the Product Owner receive it [...] and I think that this is a problem”* (06).

The real struggle, similar in all the teams, is that Marketing and Sales and other support departments are working in a separate way than the R&D department, and this is creating confusion internally but also to the customers. This creates frustration among developers, because the departments that are in a direct contact with the customers (Sales, Marketing, Customer Support for example) manage expectations in a different way: *“They should understand that part of the company is working agile, and especially the department that have direct contact with the customers should work in the same way. Because otherwise customers hear something from marketing and they are not agile, and they go to support and development and they have two different procedures, two different times of response [...] They might even look like two different companies!”* (05). This creates a problem on the customers, because if the customers interact with R&D or Sales it might seem like two diverse ways of working, which creates confusion. The developers were expressing their frustration because agile is felt by and R&D thing only, and this creates confusion in the customer when interacting with other customer-facing departments.

*“Looking overall at this, I see the same overall issue pop up, nobody else wants to be agile. People think that agile is another way that monkeys in the box are organized, but the surrounding structure, Sales, Consultancy, people around, everybody doesn’t really want to do the agile thing. If they promise a customer that something has to happen now, it has to happen now! And not in the next Sprint. [...] But in some way or another, if the rest of the organization doesn’t understand working agile, you lose a lot of effectiveness. On the other hand, with the current project, the reorganization got a lot of chaos and a lot of loss of*



*knowledge on our management layer and people with no experience of running a software department and I think that the only reason that our project managed to successfully survive was that we had a team that worked agile.” (03).*

As it was presented in the introduction, the role of the Scrum Master is to promote the agile culture within other departments. The Scrum master is responsible for the implementation of the Scrum principles within the teams, but also in the rest of the organization, and explains to the organization how to interact with the Scrum teams. Considering that Marketing and Sales and other support departments are working differently than the R&D department, the role of the Scrum Master is extremely relevant at Unit4, but this is not reflecting in practice. Because at Unit4 there is no dedicated Scrum Master role and this constitutes an issue because the other departments do not work with Scrum and this figure (Scrum Master) has a certain importance for the placement of Scrum and Scrum principles in the whole organization, not only in the R&D department. Currently at Unit4 the Scrum master is a part-time role, which is part-time Scrum master and part-time developer, in most of the cases. The role of the Scrum Master is to be the advocate of Agile, within the team, but especially in the organization. A dedicated Scrum Master role could perhaps facilitate the other departments in embracing Agile and helping customers develop realistic expectations.

In terms of the relationship with the customer *“It would help if everyone would be working agile, not just the developers, but also the consultants, sales persons, partners. That’s necessary for making agile a success and perform better to the customers.” (05).*

## 4.2 Analysis

This analysis phase is more selective. The sub-research questions will be answered, considering the relation between independent and depended variable and which structural properties are influencing the outcome variables:

- 1) *What is the relation between self-organizing potential (SOP) and the creation of customer value (CV) in R&D agile teams?*
- 2) *What is the degree by which the structural characteristics can accelerate this relation?*

Sub-question number 1 will be answered in paragraph 4.2.1 and the sub-question number 2 will be answered in paragraph 4.2.2, based on the axial results presented in the previous 4.1 paragraph.



#### 4.2.1 SOP and CV

In this paragraph the answer to the first sub-research question will be provided based on the results presented in the 4.1 axial section. In the following paragraph the results will be explicitly linked to the concepts of the theoretical framework, and the relation between self-organizing potential (SOP) and the creation of customer value (CV) will be addressed.

The degree of Scrum implementation has been analyzed after the interviews and the Scrum principles of *Transparency*, *Inspection* and *Adaptation* ((Schwaber & Sutherland 2016) entail, in their practice, working with a continuous delivery model which has a positive impact on the creation of customer value, as presented in paragraph 4.1.1. Sometimes there seems to be an overlap between concepts, because to reach a high degree of Scrum implementation, teams are indirectly working on the other indicators of the SOP. In order to achieve a high degree of Scrum implementation, teams need to have a high degree of autonomy and team features with level of intensive collaboration, with a high level of internal communication, with multi-skills and with high degree of responsiveness to change.

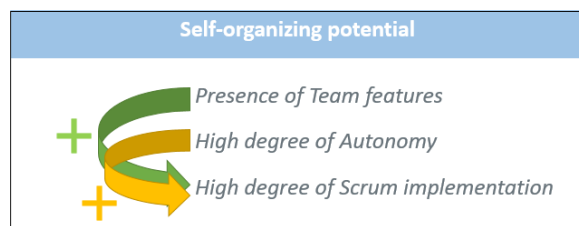


Figure 12: Team features and Autonomy reinforce Scrum implementation

As it was mentioned in the previous paragraph, working on SOP has an impact on the creation of CV, and there are significant differences in the value created for customers in teams that have embraced autonomy and Scrum principles fully, and are delivering continuously to their customers. In those teams, CV created is higher, and teams that struggle with it create lower CV. All the four teams are at least in the pilot phase with the Unit4 continuous delivery model; as it was presented in the axial paragraph, some teams have fully embraced Scrum (Team 2 and Team 3) and are working in the continuous delivery model completely. Those are the ones (Team 2 and Team 3) that are seeing the most advantage in the creation of customer value. As it was presented in 4.1.1 a high degree of Scrum implementation has a positive effect on the creation of customer value, in terms of high customer feedback implementation and high customer involvement. The implementation of Scrum principles of *Inspection*, *Adaptation* and *Transparency*

enables teams to work more easily with a continuous delivery model and to create high quality software for the customers.

The DTAP model implies high degree of *Inspection*, expressed in the Testing phase of the DTAP model, and *Adaptation*, ensuring high quality and iterative and continuous improvement of the products, via the collection of customer feedback in the Production stage of the model.

No one of the four teams is cross-functional, (the team members belong to R&D) and the implication of this finding will be explained when reflecting on the structural characteristics in paragraph 4.2.2. For now, it is very important to have multiple skills in the team as it is mentioned in paragraph 4.1.2 because the DTAP process is functioning better when there is at least one developer, one tester and one analyst in the team.

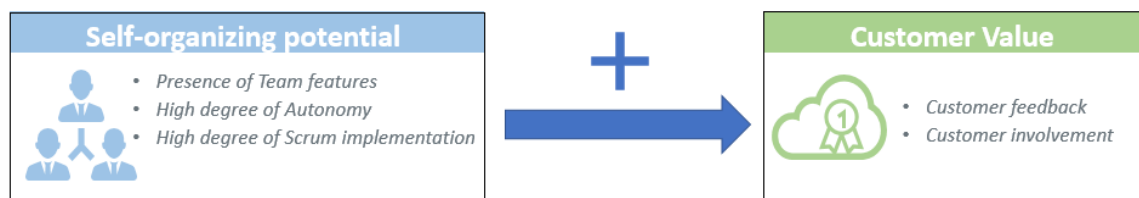


Figure 13: SOP → CV

Figure 13 summarizes what was presented in paragraph 4.1. The presence of the Team features (SOP) such as high degree of intensive collaboration, high degree of internal communication, multi-skilled teams, responsiveness to change, all combined with a high degree of Autonomy (SOP) enables the team to have a high degree of Scrum implementation (SOP) and the creation of customer value (CV). The collection of customer feedback and the involvement of the customers happen in several ways which have been presented in paragraph 4.1.1 and 4.1.2 such as the creation of users' community, *Unit4ideas* platform and via the collection of analytics in the production environment of the DTAP model.

A low degree of Scrum implementation, and a lack of autonomy have a negative impact on customer participation and customer feedback. In the 10-months release model, described in paragraph 4.1.1, it is not possible to implement all the feedback received by the customers, without throwing away the software and building it again from scratch, which is a waste of time, and has a negative impact on the implementation of customer feedback, and consequently on the customer value.

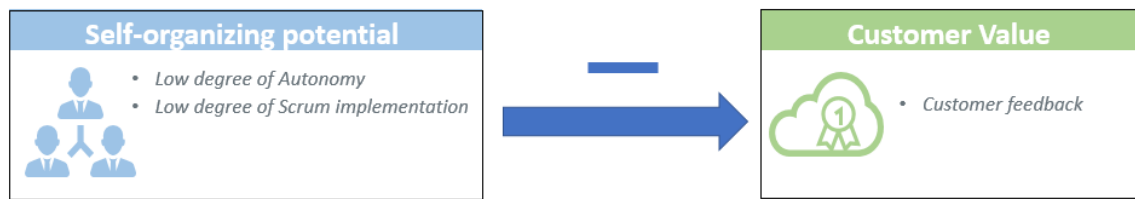


Figure 14: Negative impact on customer value, in Team 1, when dealing with a more waterfall method (10-month release)

#### 4.2.2 SOP, CV and structural characteristics

Team members see the value of being independent and to be organized with a low degree of functional concentration. A minimum level of functional concentration means that all operational tasks of different types are grouped into a “production flow” (Achterbergh & Vriens, 2009, p. 243). This is useful to avoid waiting time and proceed without being dependent on other teams to move on and quicker produce high quality software for the customers. In paragraph 4.1 a high level of independence was described for Team 1, Team 2 and Team 3; those teams have a low degree of functional concentration which is positively impacting the creation of customer value. Those teams are able to perform a complete process of continuous delivery on their own. The original relation between SOP and CV stands and it is amplified by the low degree of functional concentration. In paragraph 4.1.3 it was described that the added value of working independently and the advantage of being able to work on a full set of technology as a team. A high degree of autonomy plays a key role, and the capacity of the team to work independently. In this way the customer feedback can be implemented following the production flow and the pace of the team, without dependencies on other R&D teams. When each team is working independently on their own product this enhances the chance of the team to quickly respond to feedback, with a consequentially enhancing effect on the relation between the SOP and CV. This is visually represented in figure 15 below.

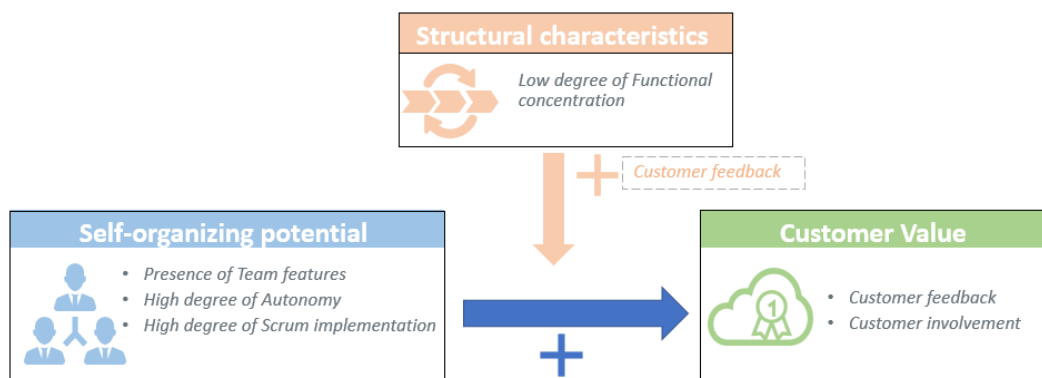


Figure 15. Positive influence of low degree of functional concentration on the conceptual model

There is an exception for Team 4: in this case the high degree of functional concentration between the back-end and the front-end teams is having a negative impact on the creation of customer value in terms of slower response to feedback in relation to the conceptual model. There is a dependency between the back-end and the front-end who need to wait for each other to deliver a product to the customer. The reason behind this choice is the necessity of creating uniformity of the back-end functionalities in several front-end products, but this is having a negative impact on the CV created when it delays the front-end team to proceed.

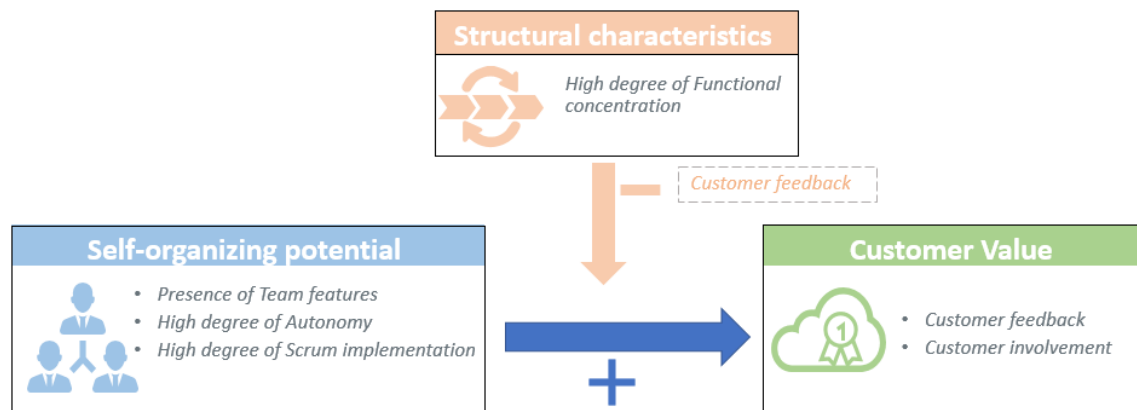


Figure 16: High degree of functional concentration has a negative impact of the conceptual model

As it was mentioned in the theoretical framework, the second parameter of De Sitter describes the level of differentiation of operational transformation. It is at an ideal minimum level if making, preparing and supporting activities are contained in operational sub-transformations (De Sitter et. Al, 1997). In paragraph 4.1.3 supporting activities have been discussed and the relation to other supporting departments (Marketing, Sales etc.) which do not work with Scrum methodology. None of the four teams is cross-functional because the team members belong to R&D. *Supporting* which includes all the activities that support the output realization, such as Sales, Marketing etc. (Achterbergh & Vriens, 2009) is external to the team. Based on the information provided in the axial analysis it can be argued that the second parameter of De Sitter is high. Every team (every product) has a dedicated support team that does first and second line support. Sales and Marketing interact directly with the customers and a lot of information is lost when reported back, which is not good for the customer. The finding that other departments are in touch with the customer, that have not implemented agile can even damage the customer relationship.

As it was mentioned in paragraph 4.1.1. there is still a lot of indirect messaging between the development Team and customers, because usually the feedback of the customer comes via Sales or Consultants or Help desk, and this causes some loss of information and the feedback is

implemented over a longer period. In figure 17, the influence of a high level of differentiation of operational transformation is represented in relation to the conceptual model, as well as the negative impact on the relation between SOP and CV.

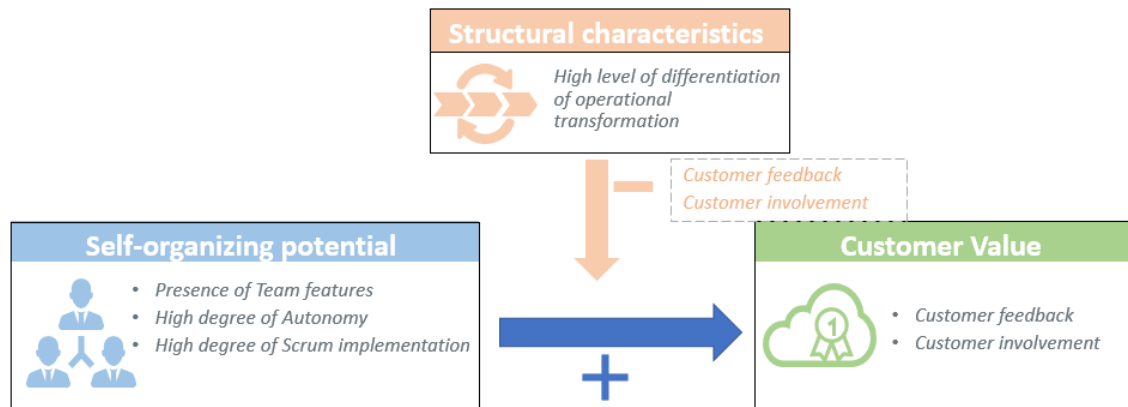


Figure 17: High level of differentiation of operational transformation has a negative influence on the conceptual model

The answer to the main research question will be presented in the next chapter, where the conclusions will be presented, taking into account the main findings so far discussed in the analysis.

## 5 Conclusion

The aim of this chapter is to summarize the findings presented in the results section and answer the main research question of this case study: *What is the relation between self-organizing potential and the creation of customer value and how is this influenced by structural characteristics, in R&D agile teams at Unit4?*

One of the main findings is that there is a high positive relation between SOP and the creation of CV. Employees improvement of self-organizing potential is resulting in an enhancement of the creation of customer value, ensuring high quality software to Unit4 customers. This relation can be accelerated by a low degree of the first structural parameter of De Sitter, and slowed down by a high degree of the second structural parameter of De Sitter. A visual representation of the conclusions is presented in figure 20 below, where the conceptual model is being pictured, including all the relevant concepts and models.

The degree of Scrum implementation is the strongest indicator of SOP in this research and it is reinforced by the presence of the other SOP concepts (team features and a high degree of autonomy). It was discovered that in order to achieve a high degree of Scrum implementation, teams need to present a high degree of Autonomy and the following Team features: elevated level of *intensive collaboration*, elevated level of *internal communication*, *multi-skilled* teams and high degree of *responsiveness to change*. A high degree of Scrum implementations means that teams work following the principles of *Inspection*, *Adaptation* and *Transparency*, while delivering continuously to their customers. In order to achieve a high implementation of Scrum principles teams need to be autonomous and have all the team features of an agile team. Multiple skills are needed within Scrum teams, such as developers, testers and analysts, to complete the release process independently and ensure a highly valuable product for the customer in the production environment. A multi skilled and autonomous team has a positive impact on the customer value and on the creation of team shared responsibility, because it has higher capacity of implementing customer feedback and involve customers in the testing phase of the software.

Working independently from other teams seems to be relevant to respond quickly and efficiently to customer feedback. That is the reason why a low degree of functional concentration at Unit4 is accelerating this relation, mostly because of a higher capacity of implementing customer feedback, which creates higher CV. One exception is discovered in Team 4, with strong dependency between the *back-end* team and the *front-end* team, which slows down the implementation of customer feedback and the team pace.

A high degree of level of differentiation of operational transformation is slowing down this relation, because of the loss during internal communication of customer feedback and the less customer involvement. Other support departments (Sales, Marketing etc.), which are customer-facing, do not work with Agile methods and this creates confusion among customers because they interact with two different working styles from developers and Sales or Marketing people, which creates discrepancies in managing expectations between customers and Unit4.

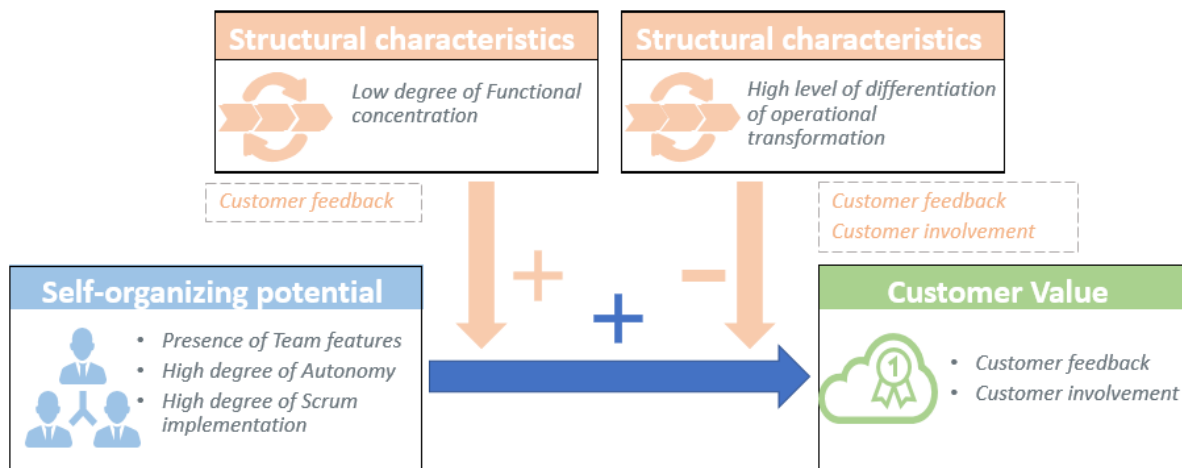


Figure 18: Final conceptual model to answer the main research question: "What is the relation between self-organizing potential and the creation of customer value and how is this influenced by structural characteristics, in R&D agile teams at Unit4?"

Overall there is a positive relationship between SOP and CV while a lack of Scrum implementation, due to holding up to a more traditional way of working software, (Team 1) is resulting in less feedback from the customers and less customer involvement. This implicates less creation of CV for Unit4 customers.

For a better and deeper understanding of the Conclusions it is highly advised to read first the Result chapter (chapter 4) where detailed explanations are provided, and relevant quotes from the interviews are presents to substantiate the findings. In the next Discussion chapter (chapter 6) managerial advice, based on the results, will be presented together with possible limitations and contribution to knowledge and future research.

## 6 Discussion

The aim of this section is to discuss the interpretations of results, the contribution to knowledge, managerial implications, and the limitations of the master thesis by means of a critical reflection and eventual directions for future research.

### 6.1 Limitations

Like all research, also this master thesis has to take into account a number of limitations that affect the quality and results of the study, by examining possible reliability and validity flaws.

There are some main points of concern, such as the *number of participants*, due to a limited time period available for conducting interviews. Validity determines whether the research truly measures what was intended to be measured or how truthful the research results are; while reliability refers to the idea of replicability or repeatability of results or observations (Cohen, Manion, & Morrison, 2013). The number of interviews can be a point of concern for validity and reliability because only seven interviews are conducted from the two principal lines of business: *SME* and *Financial Services*. This choice was made by following the advice of managers of the Benelux R&D department, which recommended Product Owner which are working closer to the customers as possible points of contacts. In order to get a full picture of the R&D Benelux organization, a future research could include interviews with all lines of business. This choice was made to get a more in-depth understanding of the teams which participated in the interviews, rather than increase the number of teams, involving one single person.

The study would have greatly benefitted from an additional observation to complete the triangulation of methods (Van de Ven, 2007). For example, beneficial would have been an observation of a Sprint review or retrospectives with a customer participation, which could not be realized because of a limited timeframe and language barrier. Most of the teams perform their Sprint reviews and meetings in Dutch, which proves to be problematic since the researcher is not a Dutch-speaker. Triangulation is typically a test strategy for enhancing the validity and reliability of research or evaluation of findings. Mathison (1988) elaborates this by saying: *“Triangulation has risen an important methodological issue in naturalistic and qualitative approaches to evaluation in order to control bias and establishing valid propositions”*. (p. 13)

*Involvement of customers*: it would have been interesting for the results of the thesis to get a validation of the results by the customers. Perhaps collecting data also from the customer point



of view and comparing the results related to the customer value to get a validation of the results. This decision was declined to avoid a too high degree of complexity in the conceptual model, which was already combining three theories. An internal perspective was chosen, but it could be an idea for future research to focus on the customer perspective and collect data from the customer perspective.

*Time of realization of the Agile project:* The Agile transformation at Unit4 is an ongoing project. This master thesis is focused on a single period in time and not on the evolution of the project. It might be interesting to conduct interviews with Development Teams and Product Owners at the end of the year, to have an idea of the evolution of the Agile transformation, which is currently moving forward.

*As an external person to the department and without a technical R&D background,* it is hard to correctly depict the complete organizational setting due to its size and complexity. Understanding the whole structure of the R&D department was a challenge due to the many different business realities which differentiates one line of business to the other, and from types of activities performed. Sometimes developers and product owners used a technical language during interviews which required investigation and further reading on the researcher side. This was interesting and challenging at the same time for the researcher. This process would actually take more time and insight than was accounted for in this research.

## 6.2 Managerial Advice

In this section the main advices for the management team will be summarized to facilitate their eventual consideration and implementation. As it was said before, since the researcher is an external person, this master thesis has no pretense to solve all the issues at Unit4, but to provide advice and bring certain topics up for discussion within the management team. Without a technical background, all the below advices have been reported by looking at the R&D department from a very novel perspective.

The biggest frustration of the R&D employees is the fact that *“Agile should not be an R&D thing only!”*. It is clear from the results and conclusion, that the non-implementation of Agile in the other support departments can create an obstacle for the creation of customer value. R&D is putting a huge effort in implementing Scrum methodology and principles, ensuring high quality products for the customers and delivering with a continuous delivery model. If R&D is left alone

with this effort, without involving Sales, Marketing and the rest of the customer-facing organization, the efforts of R&D employees could be jeopardized. Sustainable change can be achieved only when in addition to work practices the culture of Unit4 changes as well. People need to understand what is going on in the rest of the organization, facilitating an environment where it is easier to collaborate, exchange expertise, continuously improve on a global level and be better informed about the current status of all teams.

A dedicated Scrum Master would be able to promote the Scrum principles within the organization and within the teams. In this moment the Scrum Master is a part-time role, in addition to a developer/tester role. A dedicated Scrum Master role would be able to promote Scrum within the organization and to remind teams and departments of the Scrum principles, also those that are facing the customer but do not work Agile.

In the interviews the Product Owners flagged that they are working towards the involvement of the customers at the end of the Sprints. The consistent involvement of customers in the Sprints and bringing together the development teams and customers should be not negotiable and encouraged by the management team. The involvement of the customers is fundamental and should be done consistently by all R&D teams. The developers in general would like to reduce the “middle-man” between them and the customers, with the opportunity to engage in confrontation with customers more often. This will ensure to not waste precious feedback by passing on information, and will improve the value for the customers in terms of feedback implementation and direct involvement.

Another advice would be the one to bring all the teams to the same level of Scrum implementation. The new way of working leads to an ecosystem of self-organizing teams with an experimental and transparent mindset. This is based on the finding that the teams that implement Scrum completely, which are also highly autonomous and with all team features in place, are creating high customer value. Management teams should provide all the teams with the tools and resources to be able to implement Scrum completely and create high quality software for the customers. This can be challenging for certain types of products due to their maturity level and for business reality which needs to keep on working on maintenance activity.

### 6.3 Contribution to knowledge and future research

This master thesis was written considering and combining three different theories. The Agile methodology was highly considered to define the SOP and CV. Lean was included for the central role of CV and the Sociotechnical Design theory, by De Sitter, was included for an organizational

design perspective. The combination of those theories can be considered a contribution to Agile knowledge analyzed in an organizational design perspective.

This research has the aim to be of practical use for the Unit4 organization, to get insight into the R&D department and to use some of the findings in the future waves of the Agile transformation project, with an iterative and continuous way of improving.

In future research the researcher would suggest a follow-up at the end of the year at Unit4 and a validation of the results involving the customers. Future research could include the investigation of Agile in other departments which do not strictly develop software. For example, at Unit4 the HR department will start working Agile in September 2018 and this could be an interesting insight into the creation of customer value for internal customers (Unit4 employees). The extensions of working agile in other support departments is a good sign towards a more homogeneous way of working across the organization, and a first signal for other support and customer-facing departments to consider the adoption as well.

## 7 References

Cover picture: Unit4 branding library, image library (2018). Retrieved July 07, 2018, from Unit4 sharpoint site <https://unit4.sharepoint.com/sites/marketingV3>

Achterbergh, J., & Vriens, D. (2009). Chapter 7. De Sitter. In *Organizations: social systems conducting experiments*. Nijmegen: Springer, 227-280.

Blackstone, A. (2016). *Principles of Sociological Inquiry—Qualitative and Quantitative Methods*.

Butz Jr, H. E., & Goodstein, L. D. (1996). Measuring customer value: gaining the strategic advantage. *Organizational dynamics*, 24(3), 63-77.

Chow, T., & Cao, D. B. (2008). A survey study of critical success factors in agile software projects. *Journal of systems and software*, 81(6), 961-971.

Cohen, L., Manion, L., & Morrison, K. (2013). Validity and reliability. In *Research methods in education* (pp. 203-240). Routledge.

Conboy, K. (2009) Agility from First Principles: Reconstructing the Concept of Agility in ISD *Information Systems Research* 20(3), pp. 329–354, 2009.

Christopher, M., Towill, D. (2001), An integrated model for the design of agile supply chain, *International Journal of Physical Distribution & Logistics Management*, Vol. 31 Issue: 4, pp.235-246.

Christis, J. H. P., & Soepenbergh, E. (2015). Lowlands sociotechnical design theory and lean production. *Co-Creating Humane And Innovative Communities Of Work*.

De Sitter, L. U., Den Hertog, J. F., & Dankbaar, B. (1997). From complex organizations with simple jobs to simple organizations with complex jobs. *Human relations*, 50(5), 497-534.

Dixon, N. M., & Knowledge, C. (2000). How Companies Thrive By Sharing What They Know. *Harvard Business School Press*, Boston, 12(7), 8-307.

Dubois, A., & Gadde, L. E. (2002). Systematic combining: an abductive approach to case research. *Journal of business research*, 55(7), 553-560.

Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Software Development*, 9(8), 28-35.

Guzzo, R.A. and Dickson, M.W. (1996) Teams in organizations: Recent research on performance and effectiveness. *Annual Review of Psychology* 47, pp.307-338.

Hammersley, M. & Traianou, A. (2012). Ethics in Qualitative Research: *Controversies and Contexts*. Thousand Oaks: Sage

Hines, P., Holweg, M., & Rich, N. (2004). Learning to evolve: a review of contemporary lean thinking. *International journal of operations & production management*, 24(10), 994-1011.

Hu, Z., Yuang, Q., Zhang, X. (2009), Research on Agile Project Management with Scrum method, *IITA International Conference on Services Science, Management and Engineering*.

Janssen L. (2014), UNIT4 reports growth in revenues and operating profit. Retrieved May 21, 2018, from <https://www.unit4.com/blog>

Justensen, L. & Mik-Meyer, N. (2012). Qualitative Research Methods in Organizational Studies. 1st edition. Copenhagen: Hans Reitzels Forlag.

Mathison, S. (1988). Why triangulate? *Educational Researcher*, 17(2), 13-17.

Miller, T., Birch, M., Mauthner, M., & Jessop, J. (Eds.). (2012). *Ethics in qualitative research*. Sage.

Moe N.B., Dingsøy, T., Dybå T. (2008) Understanding self-organizing teams in agile software development, *19th Australian Conference on Software Engineering*, 76-85

Palmer, V. S. (2001). Inventory management KAIZEN. *In Engineering Management for Applied Technology*, 2001. EMAT 2001. Proceedings. 2nd International Workshop on (pp. 55-56). IEEE.

Schwaber, K., Sutherland, J. (2016) *The Definitive Guide to Scrum: The Rules of the Game*. Retrieved February 13, 2018, from <https://www.scrum.org/>

Takeuchi H., Nonaka I. (1986) The new product development game. *Harvard Business Review* 64(1):137–146

The 2015 State of Scrum Report (2015); Scrum Alliance. Retrieved February 20, 2018, from <https://www.scrumalliance.org>

Van de Ven, A. H. (2007). *Engaged Scholarship: A Guide for Organizational and Social Research*. Oxford, UK: Oxford University Press.

Womack, J. P., & Jones, D. T. (1996). Beyond Toyota: how to root out waste and pursue perfection. *Harvard business review*, 74(5), 140-158.

Womack, J. P., & Jones, D. T. (1997). Lean thinking—banish waste and create wealth in your corporation. *Journal of the Operational Research Society*, 48(11), 1148-1148.

Womack, J. P., & Jones, D. T. (2015). *Lean solutions: how companies and customers can create value and wealth together*. Simon and Schuster.

Westra S.J., (2017), How to stay agile and scalable in a postmodern ERP era. Retrieved February 24, 2018, from <http://www.unit4.com/blog>

Woodruff, R. B. (1997). Customer value: the next source for competitive advantage. *Journal of the academy of marketing science*, 25(2), 139.

Yin, R. K. (2017). *Case study research and applications: Design and methods*. Sage publications.

Zeithaml, V. A. (1988). Consumer perceptions of price, quality, and value: a means-end model and synthesis of evidence. *The Journal of marketing*, 2-22

## 8 Appendix

### 8.1 Appendix 1 – Two exploratory interviews on Agile and R&D department at Unit4

#### *Exploratory interview guidelines*

- 1) *The Agile transformation at Unit4 started in November 2016. How has been this journey so far? (November 2016 RFP process, February 2017 teams started to work agile)*
- 2) *How is the R&D department organized?*
- 3) *How many people are working agile at Unit4?*
- 4) *How does the Agile Expert train the teams? Is the Agile Experts and the Agile coach the same figure?*
- 5) *“Every Scrum transformation, more or less follows a predictable path along the maturity line. An Agile Coach must focus on getting the team past the red line. Employees should understand and embrace the new different team dynamic”  
What are the biggest challenges of the team?*
- 6) *You have mentioned in your previous email that the Scrum Master is a figure within Unit4? Can you explain where is this person coming from?*
- 7) *What is the relationship between Product Management, Agile and Scrum?*
- 8) *Can you explain how the Scrum Events (Sprint Planning, Daily Scrum, Sprint Review, Sprints Retrospectives) are happening at Unit4?*
- 9) *Possible contacts of people to interview (agile experts, team who are working agile)*



## 8.2 Appendix 2 – Operationalizing and Proposed interview guide

### 8.2.1 Operationalizing (examples of questions).

CONCEPTS	INDICATORS	Example of questions
<b>Self-organizing potential</b>		
<b>Degree of autonomy</b>	External autonomy	<i>How do you take decisions in relation to the rest of the R&amp;D department? Who are your main internal stakeholders and how do you interact with them?</i>
	Internal autonomy	<i>How do you take decisions around the divisions of work in the team?</i>
	Individual autonomy	<i>How do you organize your personal work?</i>
	Autonomy	<i>Who has the authority to set goals for the team? And how do you do that?</i>
<b>Team features</b>	Cross-functional	<i>How do you identify all the skills set necessary within the development team? Would you confirm that within the team you have all the skills to complete the work within the team?</i>
	Intensive collaboration	<i>How do you collaborate within your team? How do you collaborate with the rest of the organization? How do you collaborate with the customer?</i>
	Informal communication	<i>How is the communication within the team? Would you define it informal?</i>
	Responsive to change	<i>How quickly do you adapt to customer feedback? How responsive are you to external changes?</i>
<b>Degree of Scrum implementation</b>	Transparency	<i>Are all the relevant aspects of the process visible to those responsible for the outcome?</i>
	Inspection/self-assessment	<i>How do you inspect your product, without stopping the regular performance?</i>
	Adaptation	<i>How do you ensure a iterative/continuous improvement of the products?</i>
<b>Customer value</b>		
<b>Customer Feedback</b>	Feedback	<i>How do you improve based on the feedback received by the customer?</i>

<b>Customer Involvement</b>	Customer sharing of future desire	<i>How do you analyze your customer future desires? How do you implement their requests in new software features?</i>
	Performance monitoring	<i>How do you ensure the best quality of product for your customers? How do you identify wasteful activities? And how do you prevent them?</i>
	Customer relationship	<i>How did the relationship with the customer change since you started working agile?</i>
	Customer presence	<i>Do you involve the customer at the end of each sprint?</i>
<b>Structural Characteristics</b>		
<b>Functional concentration</b>	Production flow	<i>Can the team perform a complete process on its own or does it need a lot of other teams to perform this process? (low versus high FC)</i>
<b>The level of differentiation of operational transformation</b>	Integration of various tasks	<i>How R&amp;D teams interact with each other's? And other departments? Which types of operational task do you perform? Do you perform financial, HR planning activities?</i>  <i>Is the team dependent on other departments for preparing/ supporting activities?</i>

## 8.2.2 Proposed list of interviews questions for Agile teams

### **Guidelines - Interview Team 1-2-3-4.**

*Greetings. Explaining what I am doing.*

*Signing of inform consent form.*

1. *What is your role in the team?*
2. *Can you please define what are your team responsibilities? Which product/products are you developing?*
3. *How are the tasks distributed within the team?*
4. *Which types of operational task do you perform? Do you perform any financial, HR planning activities? (support activities)*
5. *Is the team dependent on other departments for preparing/ supporting activities?*
6. *Can the team perform a complete process on its own or does it need a lot of other teams to perform this process? (low versus high FC)*
7. *How is your decision-making process? Who is taking decisions?*
8. *Who has the authority to set goals for the team? And how do you do that?*
9. *How is the cooperation and communication in the team? Would you define it as informal?*
10. *Are you familiar with the 3 main pillars of Scrum? (Transparency, Adaptation and Inspection)?*
11. *Would you define your team transparent (Transparency)? (Ask for examples)*
12. *How do you ensure an iterative/continuous improvement of the products? (Adaptation)*
13. *How do you inspect your software? (Inspection)*
14. *Who are your main internal stakeholders and how do you interact with them?*
15. *How do you interact with other R&D teams in the department?*
16. *How do you improve based on the feedback received by the customer?*
17. *Would you agree that inside your team you have all the skills needed to improve the feedback of customers?*
18. *How quickly do you adapt to customer feedback? How responsive are you to external changes?*
19. *How would you define the capacities of R&D employees, in preventing and correcting errors?*
20. *How is quality control in R&D? How is your strategic product development? How quick is the innovation time?*
21. *Any input or suggestions to improve the R&D software development production-flow?*

*Do you have any questions?*

*Thank you!*

\*Keep in mind follow-up questions are very important on relations between concept.

### 8.3 Appendix 3 – Informed Consent form for interviews

This Master thesis as part of the MBA Organizational Design & Development at Radboud University Nijmegen, conducted by Federica Della Rupe. Supervisor at university is Matthijs Moorkamp.

Contact details Federica Della Rupe

E-Mail [Federica.della.rupe@unit4.com](mailto:Federica.della.rupe@unit4.com)

Phone Number +31639889615

Address: Oude Graafseweg 33, Nijmegen

An explanation of the research project was given verbally at the beginning of the interview.

Hereby I, \_\_\_\_\_ declare to be willing to participate in the interview and being recorded for transcription. I will offer you the opportunity to inspect this transcript.

For the analysis of the interview all aspects that refer back to my own person and would allow others to identify me will be anonymized or deleted. I will only be quoted in parts in the thesis so that others cannot identify me from what I said. Contact details the researcher has of me will be stored separately from the interview transcript and are made inaccessible for third parties.

I know that I am participating in the research voluntarily and have the chance, at all times, to stop the interview and refuse any further participation. Also, I can take back my agreeing to being recorded and transcribed. This will not have any negative repercussions on them from my side whatsoever.

After the thesis is completed and written I will received insight into the results of the research.

By signing this form, I agree to participate in the thesis research in form of an interview.

Signature: \_\_\_\_\_

Date of the interview: \_\_\_\_\_

## 8.4 Appendix 4 – Coding (statements cards and analysis map)

colourbar

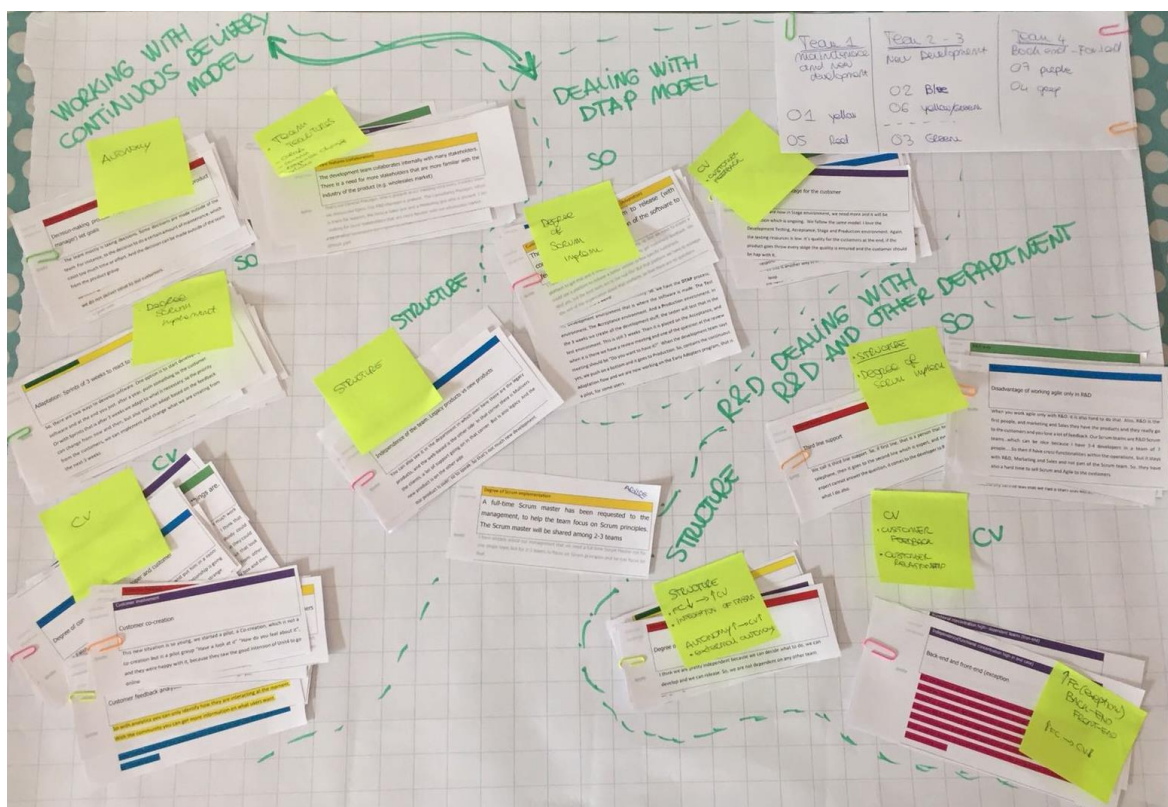
paraphrase

After printing, please mark this field in your personal colour (for easy sorting)

**BEFORE PRINTING, REPLACE THIS TEXT WITH THE PARAPHRASE: IN YOUR OWN WORDS SAY WHAT THE QUOTE MEANS**

quote

Before printing, replace this text with a quote from the transcript. Include the numbers and speakers, so the text can be found back. If you want, you can emphasize parts of the quote by making it bold.



8.4: Analysis Map (axial and selective)

## 8.5 Appendix – Scrum Roles and Unit4 R&D Benelux department

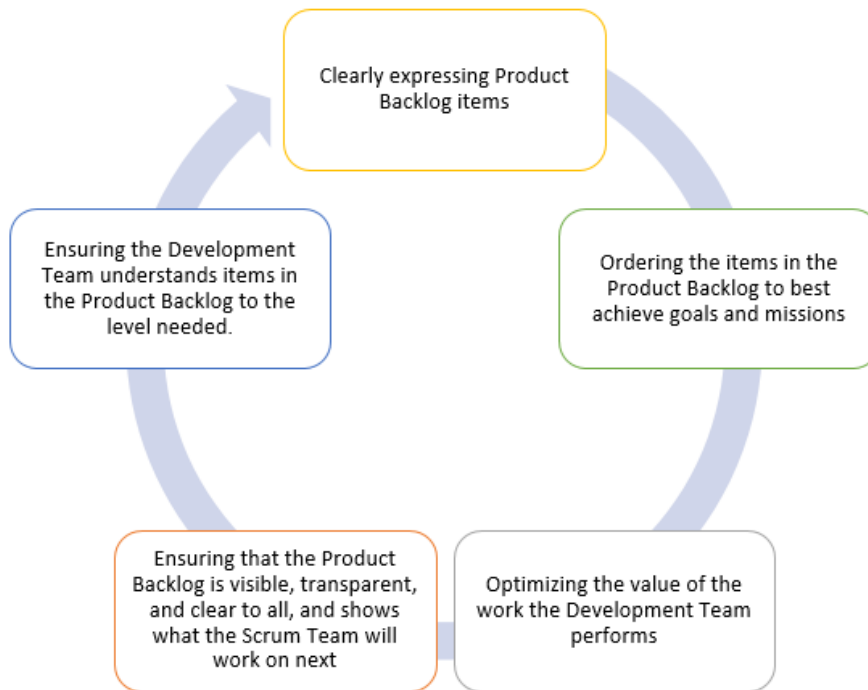


Figure 1: R&D Benelux Organisatie, Unit4 2018, Appendix version.

For a better understanding of the reader it is important to define the roles and the team structure in Scum. Scrum self-organizing teams consist of:

1. Product owner
2. Development team
3. Scrum Master

The Product Owner is responsible for the maximization of the return on investment (ROI) of the product and foster the value of the work of the development team in general. The Product owner is the only person which manage the Product Backlog, which the following.



*Appendix 8.5: Product Owner, Backlog management*

The Development Team consists in flat, self-organized team, which are responsible of creating the Increment, which is the output of each Sprint. The members organize their own work and are cross-functional, which means that all the skills necessary to create a product Increment are within the team. Even if individual team members may have specialized expertise, the whole team is accountable for turning the Product Backlog into Increment. Responsibility is shared among the whole team. The recommended team size of Development Team does not exceed the 9 members, to reduce the complexity and the interactions in order to gain productivity, and reduce high coordination (Schwaber & Sutherland 2016).

The Scrum Master is responsible of facilitate the interactions between the Scrum Team and the external teams. His or her role is to ensure that Scrum theory, practices and rules are understood and applied in the correct way. In the below figure the Scrum Master responsibilities are presented in relations with the Product Owner, with the Development Team and the Organization.



S c r u m  M a s t e r  r o l e  t o	the Product Owner	<ul style="list-style-type: none"> <li>• Finding techniques for effective Product Backlog management</li> <li>• Helping the Scrum Team understand the need for clear and concise Product Backlog items</li> <li>• Understanding product planning in an empirical environment</li> <li>• Ensuring the Product Owner knows how to arrange the Product Backlog to maximize value</li> <li>• Understanding and practicing agility</li> <li>• Facilitating Scrum events as requested or needed</li> </ul>
	the Development Team	<ul style="list-style-type: none"> <li>• Coaching the Development Team in self-organization and cross-functionality</li> <li>• Helping the Development Team to create high-value products</li> <li>• Removing impediments to the Development Team's progress</li> <li>• Facilitating Scrum events as requested or needed,</li> <li>• Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood</li> </ul>
	the Organization	<ul style="list-style-type: none"> <li>• Leading and coaching the organization in its Scrum adoption</li> <li>• Planning Scrum implementations within the organization</li> <li>• Helping employees and stakeholders understand and enact Scrum and empirical product development;</li> <li>• Causing change that increases the productivity of the Scrum Team</li> <li>• Working with other Scrum Masters to increase the effectiveness of the application of Scrum in the organization</li> </ul>

*Appendix 8.5 – Scrum Master roles in relations to: Product Owner, Development Team and Organization.*

### Scrum Development Team

- Cross-functional (e.g., includes members with testing skills, and others not traditionally called developers: business analysts, designers, domain experts, etc.)
- Self-organizing / self-managing, without externally assigned roles
- Plans one Sprint at a time with the Product Owner
- Has autonomy regarding how to develop the increment
- Intensely collaborative
- Most successful when located in one team room, particularly for the first few Sprints
- Most successful with long-term, full-time membership. Scrum moves work to a flexible learning team and avoids moving people or splitting them between teams.
- 6 ± 3 members
- Has a leadership role

### Product Owner

- Single person responsible for maximizing the return on investment (ROI) of the development effort
- Responsible for product vision
- Constantly re-prioritizes the Product Backlog, adjusting any long-term expectations such as release plans

- Final arbiter of requirements questions
- Decides whether to release
- Decides whether to continue development
- Considers stakeholder interests
- May contribute as a team member
- Has a leadership role

#### **Scrum Master**

- Works with the organization to make Scrum possible
- Ensures Scrum is understood and enacted
- Creates an environment conducive to team self-organization
- Shields the team from external interference and distractions to keep it in group flow (a.k.a. the zone)
- Promotes improved engineering practices
- Has no management authority over the team
- Helps resolve impediments
- Has a leadership role